

作战仿真模型体系构建 及应用研究

郑世明 沈寿林 吴从晖 著

電子工業出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书是一部论述作战仿真模型体系构建的专著，主要围绕作战仿真模型及模型体系内容、模型设计方法、模型组合方法和模型集成框架等展开介绍。书中采用形式化分析方法，分别对模型体系中模型的继承关系和组合关系进行建模，对模型体系的可组合性进行论证；采用基于模板的军事概念模型描述和组件化建模方法，对作战仿真模型进行设计，解决了统一建模的问题；通过服务组合方式，实现了多分辨率模型建模和基于语义的综合集成，解决了不同分辨率模型的重构和组合问题；采用面向服务的体系架构设计了模型集成框架，以服务发现、服务匹配等方式将仿真模型服务按需组合后提供给分布式仿真节点，实现模型的快速重构。

本书可供建模仿真领域的科研人员和研究生，特别是与作战仿真有关的科研院所、军事院校、部队机关等单位相关人员学习参考。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有，侵权必究。

图书在版编目（CIP）数据

作战仿真模型体系构建及应用研究 / 郑世明等著. —北京：电子工业出版社，2018.10
ISBN 978-7-121-35256-0

I. ①作… II. ①郑… III. ①作战模拟—仿真模型—研究 IV. ①E83

中国版本图书馆 CIP 数据核字（2018）第 236567 号

策划编辑：张正梅

责任编辑：刘小琳

印 刷：

装 订：

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本：720×1 000 1/16 印张：10.25 字数：197 千字

版 次：2018 年 10 月第 1 版

印 次：2018 年 10 月第 1 次印刷

定 价：78.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zltz@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：（010）88254757。

前言



随着军队信息化建设和军事训练转型的不断深入,先进的计算机仿真技术手段支持军事训练向更高层次、更大规模发展,建立完善、实用、先进、可靠的作战仿真模型体系成为一种迫切需要。作战仿真是一个复杂的过程,需要集成不同类型、不同层次、不同分辨率的模型,模型之间存在继承、组合、交互和指挥控制等复杂关系,如何构建科学、合理的作战仿真模型体系是建模与仿真面临的一个重要难题。针对这一问题,本书采用形式化分析方法、组件化建模方法、基于模板的军事概念模型描述方法和面向服务的柔性体系集成框架设计方法,主要围绕模型及模型体系内容、模型设计方法、模型组合方法和面向服务的仿真模型集成框架等一些关键问题展开介绍,实现模型的统一管理、快速重构和灵活组合。

全书分为7章,主要内容包括需求研究、理论研究、技术研究、方法研究和应用研究。

第1章绪论,介绍研究背景、意义和现状,明确主要研究内容和组织结构。

第2章从基本概念、模型构建要求、作战仿真模型体系及其应用等方面对作战仿真建模的基本问题进行分析。

第3章通过梳理模型体系构建存在的问题,对作战仿真模型体系进行形式化建模分析,采用ISM方法对组合关系和继承关系进行建模,为构建可组合的模型体系奠定理论基础。

第4章研究基于组件的模型设计方法。通过模板的方法对军事概念模型进行描述,在此基础上按照组件化建模的方法对陆军作战仿真模型进行设计,并构建相应的模型体系。

第5章研究面向服务的模型组合方法。通过对模型服务描述,按照服务组合的规范,构建模型服务匹配模型、模型组合语法、模型组合规则、组合算法,并采用一种链式服务组合建模方法对模型组合服务建模,为模型组合提供方法指导。

第6章主要对面向服务的集成框架设计与实现进行研究。通过平台无关的通用协议进行服务调用,按照服务规范对仿真模型服务进行描述并发布,通过服务发现和服务选择动态构建满足实际需求的组合服务,以满足不同需求的仿真应用。

第7章主要从完成的工作和实现的创新等方面对书中的内容进行总结,并探讨和分析了需要进一步研究的问题,为其他学着重进行深化研究提供参考。

在撰写过程中,作者参考或引用了国内外一些文献成果的有关论述,在此向这些文献的作者表示感谢。感谢长期从事作战仿真研究的李路遥教授、侯俊教授、王洪军教授、韩冰副教授、张国宁副教授、谭玉玺副教授、杜国红副教授及孙鹏讲师,在本书的撰写过程中他们给予了大力支持和帮助。另外,高志年、王智新、张永亮、徐顺福、任在安、李欣对本书做出大量的审校工作,并提出宝贵意见。

由于作者水平有限,书中难免存在不足之处,希望广大读者批评指正。

作 者

2018年8月

目 录



第 1 章 绪论	1
1.1 研究背景	1
1.2 研究意义和目的	2
1.3 国内外研究现状	3
1.3.1 美军研究现状	3
1.3.2 我军研究现状	5
1.4 研究内容	7
1.4.1 研究对象及范围界定	7
1.4.2 主要研究内容	8
第 2 章 作战仿真建模的基本问题	12
2.1 相关概念	12
2.1.1 作战仿真	12
2.1.2 作战仿真模型	13
2.1.3 仿真模型体系	14
2.2 仿真模型构建要求	15
2.2.1 可信性	15
2.2.2 重用性	16
2.2.3 可扩展性	16
2.2.4 易维护性	17
2.3 作战仿真模型体系	17
2.3.1 要素组成	18

2.3.2	体系内容·····	20
2.3.3	设计方法·····	21
2.4	作战仿真模型体系应用·····	23
2.4.1	模型共享·····	23
2.4.2	模型服务·····	24
2.4.3	组合应用·····	25
2.5	本章小结·····	26
第3章	作战仿真模型体系分析与建模·····	27
3.1	仿真模型军事需求分析·····	27
3.1.1	模型特点分析·····	27
3.1.2	模型目标分析·····	29
3.1.3	组件化建模需求分析·····	30
3.1.4	组件化建模作用分析·····	31
3.2	模型体系构建分析·····	32
3.2.1	模型体系构建要求及原则·····	32
3.2.2	模型体系构建所面临的问题·····	34
3.2.3	模型体系组合性分析·····	35
3.3	作战仿真模型体系形式化建模分析·····	38
3.3.1	基本概念·····	38
3.3.2	ISM方法与步骤·····	39
3.3.3	继承关系建模·····	40
3.3.4	组合关系建模·····	43
3.4	可组合的模型体系层次结构分析·····	47
3.5	本章小结·····	50
第4章	基于组件的作战仿真模型及体系设计·····	51
4.1	军事概念模型设计·····	51
4.1.1	军事概念模型建模思路·····	52
4.1.2	基于模板的概念模型描述与设计·····	53
4.2	作战仿真模型组件设计·····	60
4.2.1	组件设计原则与要求·····	61
4.2.2	组件描述·····	61

4.2.3	组件模板与视图设计	65
4.2.4	组件开发流程	69
4.2.5	模型组件架构	71
4.2.6	组件接口设计	78
4.2.7	组件装配	79
4.3	模型通信及其消息管理机制	82
4.3.1	基于消息的通信方法	83
4.3.2	基于控制指令的通信方法	84
4.4	作战仿真模型体系结构及内容	86
4.4.1	模型体系需求分析	87
4.4.2	模型体系框架设计	87
4.4.3	模型体系内容	89
4.5	本章小结	92
第 5 章	面向服务的作战仿真模型组合方法	93
5.1	模型组合的基本概念	93
5.2	模型服务描述	94
5.2.1	原子服务	95
5.2.2	基本服务	95
5.2.3	复杂服务	96
5.2.4	组合服务	98
5.3	面向服务的仿真模型组合建模方法	99
5.3.1	组合建模需求	99
5.3.2	服务组合方法	100
5.3.3	服务组合语法	101
5.3.4	服务组合规则	102
5.3.5	服务组合算法	104
5.4	服务组合示例	107
5.4.1	仿真服务基本需求	107
5.4.2	模型组件设计分析	107
5.4.3	模型组合建模	108
5.4.4	链式服务组合建模	109
5.5	本章小结	112

第 6 章 面向服务的模型集成框架设计与实现	113
6.1 需求分析	114
6.2 基于 SOA 的集成框架分析	116
6.3 基于 SOA 的仿真模型集成框架总体设计	121
6.3.1 资源层	122
6.3.2 构件层	123
6.3.3 服务层	123
6.3.4 应用层	126
6.4 仿真模型集成框架功能设计	128
6.4.1 模型管理功能	128
6.4.2 模型封装功能	133
6.4.3 模型组合功能	136
6.4.4 模型集成应用功能	140
6.5 本章小结	142
第 7 章 总结与展望	143
7.1 研究工作总结	143
7.2 需进一步深入研究的问题	144
参考文献	147

缩略词表



WSDL (Web Services Description Language)	Web 服务描述语言
XML (Extensible Markup Language)	可扩展标记语言
UDDI (Universal Description Discovery and Integration)	统一描述、发现和集成
SOAP (Simple Object Access Protocol)	简单对象访问协议
URI (Uniform Resource Identifier)	统一资源标识
MSMP (Modeling and Simulation Main Plan)	建模仿真主计划
JWARS (Joint Warfare System)	联合作战系统
BSE (Battle Space Entity)	战场空间实体
JWSIMS (Joint Warfare Simulation System)	联合作战仿真系统
JCMMS (Joint Conceptual Model of Mission System)	联合使命空间概念模型体系
MMF (Military Modeling Framework)	军事建模框架
FLAMES (FLexible Analysis Modeling and Exercise System)	柔性分析建模和 演习系统
ModSAF (Modular Semi-Automated Force)	模块化半自动兵力系统
EBI (Entity、Behavior、Interaction)	实体、行为和交互
MCMT (Military Concept Model based on Template)	军事概念模型建模方法
SMM (Service Matching Model)	服务匹配模型
ISM (Interpretive Structural Model)	解释结构模型
BEWSC (Based-Events Web Services Composition)	基于事件的 Web 服务组合
MERA (Mission-Event-Restriction-Action)	任务—事件—条件—行动
SOA (Service-Oriented Architecture)	面向服务的架构
IFSM (Integrated Framework of Simulation Model)	仿真模型集成框架

HLA（High Level Architecture）高层体系架构

DCMF（Defense Conceptual Modeling Framework）作战概念建模框架

DAWS（Discovery Algorithm of Web Service）Web 服务发现算法

MDA（Model Driven Architecture）模型驱动架构

XMSF（Extensible Modeling & Simulation Framework）可扩展建模仿真体系框架

CMMS（Conceptual Modeling of Mission Space）任务空间概念模型

UML（Uniform Modeling Language）统一建模语言

MSRR（Modeling and Simulation Resource Repository）建模与仿真资源库



1.1 研究背景

本书针对陆军作战仿真模型体系构建、开发、重用与管理等方面存在的不足展开研究，目的是：①进一步规范化、标准化陆军作战仿真模型描述与设计方法；②建立一批适应陆军战斗力生成模式转变、满足陆军部队装备论证、作战概念创新、战法实验和作战指挥训练等需求的新一代作战仿真模型；③以陆军部队为基本研究对象，构建作战仿真模型体系，实现仿真模型的开发、存储与资源共享一体化，将分散的军事模型统一管理、科学管理、规范管理，通过服务组合的方式最大限度地实现模型重用，避免重复投资、重复建设，提高模型的可信度和互操作性。

随着军队信息化建设和军事训练转型的不断深入，以先进的计算机仿真技术手段支持军事训练向更高层次、更大规模发展，建立完善、实用、先进、可靠的作战仿真模型体系成为一种迫切需要。当建模方法发展到一定程度时，为适应其发展，需要把作战仿真模型建设作为体系工程来抓，解决困扰建模发展的技术瓶颈问题，从而支持信息化条件下联合作战指挥与训练，为军事斗争准备和部队信息化建设发挥更大作用。目前面临的问题主要来自以下两个方面。

(1) 模型质量制约着信息系统的发展。模型是训练信息系统、指挥信息系统和作战实验系统的核心。在军事信息系统研制中,长期制约其发展的是军事模型的质量。传统模型的可信度不高是核心问题之一,其原因主要来自三方面:①模型本身不具有开放性。军事人员和系统用户只能通过系统应用功能间接了解模型的合理性,仿真模型与现实有比较大的差距,导致模型描述规则和算法的合理性无法得到军事人员和系统用户的认可。②模型的自适应性弱。军事人员和系统用户通常看到的是基础数据、想定数据和模型运行的中间数据和结果数据,最核心的作战行动数据被长期封装在捆绑式的模型系统黑匣子里,模型结果的科学性、可靠性和合理性得不到监督,发现问题时又不能及时更正。③模型可信度渐弱。军事模型一旦投入应用,军事人员和模型用户不能直接与模型打交道,由此造成最应该适应变化的仿真模型无法及时修正与更新,有的模型已经严重滞后于军事和技术的发展,导致模型系统用得越久,其可信度越低,从而使军事模型不能用、不敢用。

(2) 传统建模方法制约信息系统的发展。在传统建模中,从概念建模、数学建模到程序建模,都将抽象描述与行动数据捆绑在一起。建模速度快,容易完成模型和模型系统的构建,但这样构建的模型,利用率低,很难在其他系统中重用;在建模过程中或建模之后,对模型修改与更新困难。如果要改变作战行动模型中的某一要素、量纲、指标值或元数据,就必须分别修改概念模型、数学模型、程序模型中对应的参数,不仅造成重复工作,而且会产生多道工作程序,难以保证同一模型描述内容的一致性。

1.2 研究意义和目的

开展陆军作战仿真模型体系研究,其意义和目的在于:

一是为进行陆军部队作战仿真提供模型基础和依据。在新型陆军部队建设过程中,一方面,需要检验陆军部队现有信息化装备的作战能力,验证是否达到预期的作战要求;另一方面,需要提出对未来陆军部队装备建设的新需求。构建新型陆军作战仿真模型体系,是运用作战仿真实验方法检验陆军部队作战能力的基础性工作。构建作战仿真模型,能够对陆军部队进行从局部到整体的渐进式实验,不断发现陆军部队在编制体制、作战运用模式及武器装备体系编配等方面存在的

问题,使之及时调整、修改和完善,确保陆军部队建设的科学合理。

二是采取聚合与组合相结合的方式构建陆军作战仿真模型体系,规范模型接口,以组件化的方式对战场空间实体进行抽象和分类,便于搭建一个灵活、可扩展的模型体系框架,实现模型重用与组合。能根据作战实验的实际需求,快速构建新的仿真模型,提高仿真模型资源的利用效率。在面向重用、组合的模型体系框架支撑下,节约模型开发成本,提高开发效率,缩短开发周期,提升建模仿真的灵活性和可扩展性。

三是提高仿真模型的规范性和可信性。按照组件化的设计思想,使军事现实空间与仿真空间之间保持良好的一致性,在模型描述、模型交互和模型应用等方面进行规范和统一,使其成体系化、规范化,具有可集成、可互联、可扩展的特点,提高仿真结果的可信度。以陆军作战仿真模型为典型代表,为进一步探索涉及各军兵种、跨领域、跨部门的大型复杂仿真系统的模型开发提供方法指导和论证,最终形成具有横向联合各军兵种、纵向实现各作战层次的多粒度模型体系,为装备论证、作战概念创新、战法实验、作战仿真训练等提供模型支撑。

1.3 国内外研究现状

目前,国内外对于仿真模型体系设计方法主要还是以面向对象建模方法、组件化设计方法和面向实体的方法为主,以“组件”“单元”“平台”“战场空间实体”等为单位,根据不同功能、按照不同粒度描述仿真模型,通过继承、多态和聚合等方法描述模型间的关系并实现模型的组合与重用。对于模型体系的描述和设计缺少具体的形式化方法,不同应用领域的人员缺乏模型体系设计的共同标准和规范。此外,这些设计方法主要描述了模型之间的继承、组合关系,缺少对模型体系整体结构的分析及其他关系的描述,使模型体系在多个方面存在不足。

1.3.1 美军研究现状

美军在作战仿真领域的研究领先于其他国家,也代表了军事建模仿真的较高水平。在美军建模与仿真办公室 DMSO (Defense Modeling and Simulation Office)

的建模仿真主计划 MSMP (Modeling and Simulation Main Plan) 的指导下^[1], Cotton 和 Dudgeon 分别提出开发标准的单元层对象模型和平台层军事对象模型的仿真模型体系设计方法, 并使用这两种方法对美军已有仿真系统的模型进行了全面分析, 提出了模型体系的基本概念^[2]。开发标准的平台层军事对象模型的体系设计方法, 指出模型之间的互操作性和重用性不高是因为缺少一个共同的技术框架指导仿真模型的开发^[3]。该方法从平台的角度对美军的几个仿真系统模型体系进行分析, 通过面向对象建模方法描述军事模型的属性及其继承关系, 并采用组件化的方法构造可重用的仿真模型。下面以美军几个典型的作战仿真系统为例, 介绍其模型体系的组成及构建方法^[4]。

联合作战系统 JWARS (Joint Warfare System) 是面向战役级联合作战的仿真系统。JWARS 的模型体系是单元层模型体系的一种, 主要包括战略级的后勤保障模型、战役级的后勤保障模型及各军兵种的联合作战模型, 以模型组合的方式构建战场空间实体 BSE (Battle Space Entity), 其核心是强调模型组合能力, 通过模型组合实现模型重用。采用面向对象的建模方法对战场空间进行分析和描述, 利用多态和继承等方法描述 BSE 的组合关系。

联合作战仿真系统 JSIMS (Joint Simulation System) 是面向 21 世纪联合作战的仿真系统。JSIMS 的模型体系主要是由联合使命空间概念模型 JCMMS (Joint Conceptual Model of Mission Model) 构成的, 采用军事建模框架 MMF (Military Modeling Framework) 的方法设计其结构和模型。该框架支持集成和多分辨率模型的建模, 主要目的在于对 JSIMS 中所涉及的各种模型进行有效的管理, 以支持模型的重用和组合。此模型体系的特点在于: 一是提供了一套军事领域的基础类, 使模型开发者能建立专用模型, 为模型设计和开发提供了标准模板, 为模型重用提供了可能。二是提供了一种灵活的框架, 使模型开发者同仿真构造的细节分离开, 支持新模型的自动生成, 支持模型组件间的自动重组。三是利用了面向对象建模中继承、多态和聚合等方法设计模型体系, 采用统一的、连续的、可重组的方式建立任务空间对象, 实现了模型的组合和重用。

柔性分析建模和演习系统 FLAMES (Flexible Analysis Modeling and Exercise System) 是一个商业化的开放体系的仿真框架, 主要是实现作战行动仿真开发和应用, 对多种类型的系统进行行为建模。其特点是容易使用、开放体系和高度定制化, 采用面向对象和面向过程的多种建模方法, 模型体系通过定义一个较大的基类集合完成, 这些基类由环境模型、装备模型、消息模型和认知模型等组成,

所有其他的模型从这些基类中继承获得。将运行时的模型称为单元,单元由装备模型和认知模型组合而成,在环境模型和消息模型的共同作用下进行各种仿真活动。装备模型负责模拟仿真武器装备的功能和交互过程,认知模型模拟人的认知行为和决策过程。

美国陆军下一代一体化的 CGF 仿真系统 OneSAF (One Semi-Automated Force) 及其模型框架,按照组合—产品—组件的体系模式定义其模型体系结构,按照组件化的设计思想,通过统一建模语言 UML 揭示系统组合、产品和组件之间的聚合关系,支持多层次、多分辨率模型的组合。通过物理智能体组件、行为智能体组件和原子行为组件完成对实体模型、单元模型和组合行为模型的建模,整个模型体系具有灵活性、可扩展性、集成性和实用性等特点。基于元数据描述和编程语言的动态反射机制构建如信息服务、仿真对象数据库、物理模型和行为模型的组合框架等。

可扩展建模仿真体系框架 XMSF (Extensible Modeling & Simulation Framework) 是美国海军研究生学院与 SAIC 公司等机构提出并启动的一个建模框架。XMSF 定义为一组基于 Web 的建模与仿真的标准、描述及推荐准则的集合,其核心是采用通用的技术、标准和开放的体系结构促进建模与仿真应用在更大范围的互操作性和重用性。

1995 年,美国国防部建模与仿真办公室发布了《联合作战仿真对象库》JWSOL (Joint Warfare Simulation Object Library) 规范, JWSOL 运用面向对象分析与设计的方法对战役级联合作战进行建模,是一个可靠的、可重用的作战仿真模型库和对象知识库,支持联合作战模型和想定的开发与设计,对模型的移植、互操作及重用具有很好的支持作用。

1.3.2 我军研究现状

随着计算机技术、仿真技术和信息技术的飞速发展,我军广大科研人员研制开发了大量的作战仿真模型,由此带来的模型开发费用、开发效率、开发质量、模型可信性及可靠性的控制等问题十分突出。对相同或相似的军事问题,研发人员做了大量重复的工作,无法或很少重用已有的成果,模型的开发始终不能摆脱手工作坊的方式。

目前,国内已经进行了大量卓有成效的建模与仿真研究工作,而且也初步建立了一些具有代表性的仿真系统。各军兵种针对自身的实际,开发了大量的作战仿真模型,并各自形成了相对独立的模型体系,但其中仅有部分军事概念模型按照统一的模板进行设计,因此导致用于作战仿真的各种数学模型、程序模型缺乏统一的规范,使得大部分的模型没有能够实现共享。在需要其他军兵种模型时往往需要重新设计和开发,严重影响到建模与仿真的互操作性、可重用性和可信性。

我军对模型体系的重视始于一系列军事仿真工程,通过工程推进和系统建设,积累了大量的军事仿真模型,但各军种之间的模型并没有按照统一的标准和接口进行开发,因此模型之间的交互均是按照自己的运行机制进行访问,给多军兵种的联合作战仿真实验带来了不少麻烦。很多单位主要是以跟踪美军的仿真模型体系设计方法为主进行相关的研究。例如,针对美军的任务空间功能描述(Functional Description of Mission Space, FDMS),国内也有不少的单位跟进研究,如军事科学院、国防科技大学及装甲兵工程学院等^[5]。国防大学最早展开对联合作战仿真模型体系的研究,文献[6]阐明了联合作战模拟中武器装备体系的层次性、组合性和模型之间的信息控制关系等。文献[7]从模型工程的角度探讨了联合作战仿真模型体系的设计方法,提出了模型体系的设计目标、标准与原则和基于模板的模型设计方法。文献[8]对空间系统军事应用仿真的模型进行了功能上的分类,设计了空间系统军事应用仿真模型体系结构。文献[9]提出了面向联合作战仿真任务的装备保障仿真模型体系结构,从功能角度对模型进行分类,分析了模型之间的关系。文献[10]以仿真模型重用为目标,研究了作战仿真模型体系的结构、模型分类及模型层次关系,提出了通用的仿真模型集成框架。此外,还有海军工程大学、空军工程大学、陆军指挥学院和陆军步兵学院等单位也对作战仿真模型体系的设计做了研究。

文献[11]使用基于组件(构件)的软件复用技术,可以将大量已有的仿真资源抽象、设计并成为一个可重用的组件,并建立仿真模型组件库,来组织、存储和管理可重用组件。

文献[12-14]指出在全军范围内建设统一的作战仿真模型体系的必要性,将软件模型划分为单元模型、基本模型、组合模型、联邦成员模型和联邦模型5个层次功能。文献[15-18]研究了基于决策支持系统的模型体系,介绍了模型、数据库

及其管理系统、模型库及其管理系统、方法库及其管理系统的关系和实现技术。文献[19,20]采用面向对象的方法实现了对模型体系的设计和管理。文献[21]对模型库的重用理论、方法和异构集成技术进行了研究,增强了仿真应用的标准化和互操作性。

随着仿真技术的不断发展和仿真规模的日益扩大,实现模型资源的可重用变得越来越迫切,仿真模型体系及其管理系统的需求也越来越明确。国内已有多家院校对仿真模型体系及其管理系统进行研究^[22]。北京航空航天大学主要对飞行动力学仿真模型体系进行了研究和设计,形成了能够完成各种航空航天飞行的模拟系统。哈尔滨工业大学对建模与仿真资源库(Modeling and Simulation Resource Repository, MSRR)做了一定的研究,提出了符合组件化思想的MSRR概念模型体系。国防科技大学提出了以仿真模型体系为核心建立分布式的仿真应用系统。

文献[23]对模型体系的组合性进行分析,提出面向组合的模型体系设计方法,设计了模型体系的可组合的分层结构模型并对其进行形式化分析;结合工程实践,指出模型语法组合的不足,提出了基于领域规则的模型组合方法。

综上分析,我军对作战仿真模型体系的研究还缺乏统一的顶层设计、统一标准和技术规范。

1.4 研究内容

1.4.1 研究对象及范围界定

本书以信息化条件下联合作战为背景,针对陆军部队作战特点和实际战场环境条件,以陆军部队所担负的作战任务为依托,以主要的作战样式为支撑,在一定的作战编成、兵力部署条件下,以陆军部队典型的编制体制和主要武器装备为基础,以陆军部队的能力构成、层次结构及其相互关系为主要建模对象,对陆军部队的各种实体的属性、行为和关系进行需求建模、概念建模和数学建模。在此基础上,按照一体化建模的规范化、标准化的要求,生成陆军作战仿真模型组件,

以陆军作战仿真模型为构建重点,采用面向实体和组件化建模方法,建立和开发实用、可靠、易于调用和重用的作战仿真模型资源库,为实现陆军作战仿真实验提供模型支持。

本书主要从技术角度解决仿真模型体系及构建方法问题,而这种方法是一种通用的方法,不仅是针对陆军部队的专用模型体系构建方法,还有一种能满足其他作战仿真应用的通用模型体系构建方法,能够解决军事建模与仿真领域的多样性、不同类型部队作战建模仿真的差异性等方面问题。

1.4.2 主要研究内容

着眼一体化建模技术的创新,研究陆军作战仿真模型的建模方法以提高模型资源建设水平,使规则、模型、方法和数据相互独立,便于建模人员和军事人员不受建模过程约束,以适应军事训练方式、军事技术的变化与发展需要,增强模型库的通用性、可靠性和可维护性。作战仿真模型体系研究是一项系统复杂的工作,本书围绕仿真模型体系分析、设计、组合和集成框架等一些关键问题展开研究。主要从技术与方法角度对模型体系进行研究,首先是模型体系的形式化分析与建模方法研究;其次是基于组件技术的模型设计与体系设计方法研究,在可扩展的模型体系下研究面向服务的模型组合方法;最后对模型集成框架的技术与设计方法进行研究。本书组织结构如图 1-1 所示。

研究内容分为以下五个方面。

1. 作战仿真建模的基本问题

研究陆军作战仿真模型问题,必须首先弄清与研究对象相关的基本概念,从作战仿真建模的基本概念入手,阐明作战仿真、作战仿真模型及其仿真模型体系的概念及相关关系,提出仿真模型构建的基本要求,并围绕仿真模型体系的要素组成、内容构成、设计方法及其应用问题进行分析,这既是研究的基础,也是组件模型研究的基础和仿真模型体系研究的基础。

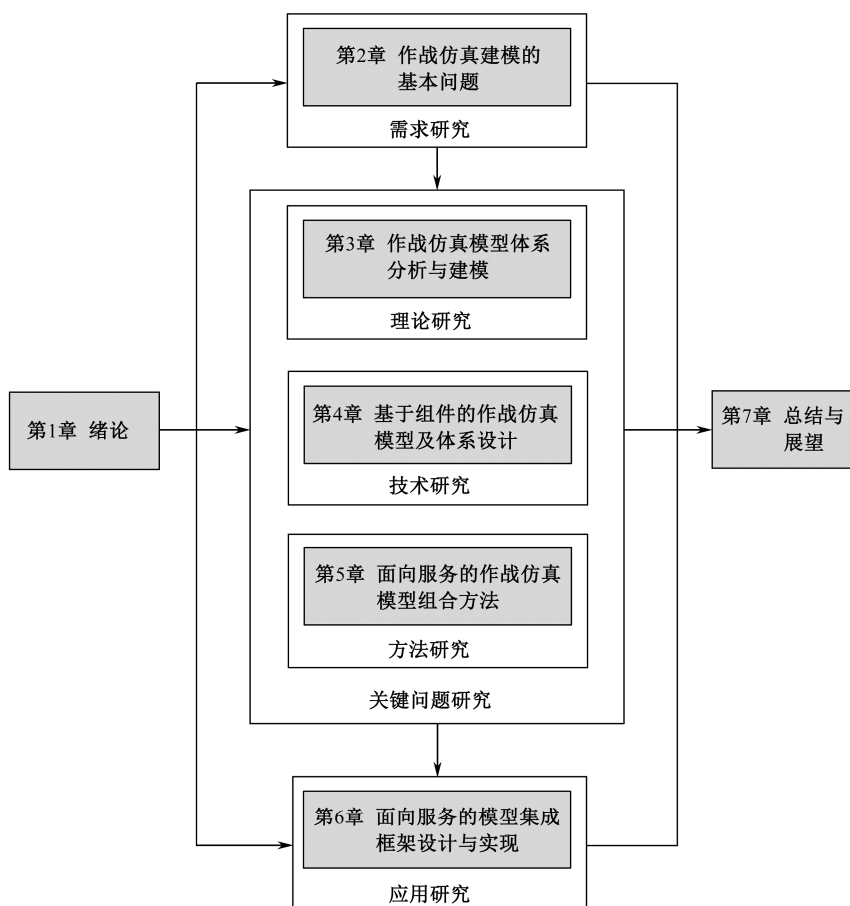


图 1-1 本书内容组织结构

2. 陆军作战仿真模型体系分析与设计

为了满足陆军部队开展多种样式作战仿真实验及方案论证的需要，必须为陆军作战仿真实验系统提供功能齐全、内容完整、科学合理的仿真模型资源。在本书中，主要针对陆军师、旅、团、营、连、排等实体的作战行动在不同层次的仿真需求构建其相应的模型体系。模型体系的研究主要包括以下五个方面：

- 作战仿真模型体系的构成要素
- 作战仿真模型的分类方法

- 作战仿真模型体系层次结构
- 作战仿真模型体系形式化建模分析
- 可组合的模型体系层次结构分析

3. 基于组件的仿真模型及体系设计研究

仿真模型的组件化设计是仿真模型重用的重要方法,在组件化建模思想的指导下实现模型的灵活重构和组合,以实现仿真模型更小粒度的重用。为解决模型的兼容性问题、跨平台重用、异构模型交互及统一访问问题,主要围绕以下六个方面对组件化建模进行研究:

- 基于组件化思想的军事概念模型描述与表示
- 组件化建模的流程
- 模型组件架构
- 组件模型接口设计
- 组件模型之间的通信方法
- 面向组件的仿真模型体系及内容

4. 面向服务的作战仿真模型组合方法研究

模型服务的目标是作为作战仿真系统应用提供统一的军事模型规范,不仅为模型的使用者提供一个共享存储与共享运行的平台,而且也为模型的开发者提供标准规范,减少模型开发过程的无序性。服务具有组合的功能,通过服务组合可以实现更为复杂的服务,将每个原子模型看成一个服务,执行组合模型实际上就是执行多个原子模型的一种特定组合,通过服务组合方式实现多分辨率模型建模和基于语义的综合集成,解决不同分辨率模型的重构、重用问题,实现多分辨率模型的灵活调用控制。本书将重点研究以下四个方面的问题:

- 模型服务表示和描述问题
- 模型服务匹配问题
- 服务组合规则问题
- 服务组合算法问题

5. 面向服务的仿真模型集成框架研究

采用服务的方式实现模型的集成应用的目的在于提高仿真模型的重用性和

组合性，将功能独立的模型组件封装成标准的模型服务，通过服务发现和服务选择动态构建满足实际需求的更为复杂的组合服务，以实现模型的快速重构和柔性组合。本书将重点研究以下三个方面内容：

- 框架功能需求分析
- 基于 SOA 的模型集成框架总体结构设计
- 基于 SOA 的模型集成框架的功能设计

第 2 章

作战仿真建模的基本问题



作战仿真建模是按照一定的军事目的和要求对各类军事对象进行抽象描述的过程。构建模型是进行作战仿真的基础。本章主要从作战仿真建模的基本概念入手，阐明作战仿真、作战仿真模型及作战仿真模型体系的概念及相互关系。从可信性、重用性、扩展性及易维护性等方面提出了仿真模型构建的基本要求。同时，对仿真模型体系的要素组成、内容构成、设计方法及其应用问题进行了分析。

2.1 相关概念

2.1.1 作战仿真

作战仿真是针对军事领域的一种仿真应用形式，主要是运用仿真技术与方法实现对作战问题的系统性研究。作战仿真是以计算机仿真为基础，主要采用系统工程、军事运筹学、信息技术等理论与方法，用物理仿真、数学仿真等方法研究战争的组织指挥、作战训练，从本质上对遂行作战任务的实体行为特征、规律进行抽象，再用物理仿真、逻辑思维或数学表达式对其进行描述的一种形式，从而

揭示作战训练活动规律。通过作战仿真能够反映作战双方兵力兵器运用、作战行动的规律及它们之间的物理和信息联系。作战仿真是从实际作战训练问题空间到虚拟作战训练问题空间转换的重要纽带,其主要任务就是构建模型、采集数据和建立规则。

从仿真对象与内容上看,作战仿真主要包括战场环境仿真、作战实体仿真、指挥控制仿真、作战行动仿真、武器装备仿真及作战效能仿真等。其中,战场环境仿真主要是对作战行动及与指挥活动相关的地理环境、战场目标、气象水文、电磁环境、近地空间等战场环境因素进行抽象描述。作战行动仿真主要是以作战行动实体为描述对象,着重对实体的属性、任务类型、战斗动作及行动规则等进行抽象和描述,建立实体与战场环境间及与其他实体间的相互关系。指挥控制仿真主要是以各级指挥实体为描述对象,着重对实体的属性、实体指挥活动及其外部形态、行动控制的内容与规则进行抽象描述。

从仿真形式上看,既有物理仿真也有数学仿真,既有实兵仿真、虚拟仿真也有构造仿真等。虚拟仿真是指人操作模拟器,人的行为由真实的人完成,作战场景、音响和作战效果通过仿真的方式产生,提供给人的是逼真的、战斗威胁十足的作战环境。构造仿真则是由虚拟的人操作仿真的系统,典型的就是“人不在回路”的仿真推演。

从仿真应用领域看,作战仿真系统可以用于作战方案推演与评估,实现对作战过程的全维仿真,为动态调整作战计划提供依据。可以用于军事训练,通过仿真手段可以在一定程度上解决过程不可复现、未来不能预演的问题,支撑各级指挥员及其机关练指挥、练战术、练谋略、练流程的训练需要,而且这种训练可以在同一作战背景、同一初始条件、同一作战对手的情况下进行反复训练,以不断提高训练的实用性和针对性。

2.1.2 作战仿真模型

作战仿真模型是以作战体系为研究对象建立的各种模型,是对作战过程的抽象和描述,应能反映现实军事系统或现象的基本特征。作战仿真模型是对作战过程中的战场环境、各类实体、作战行动、交战过程和毁伤结果的具体抽象,构建作战仿真模型的目的是采用定量分析的方法研究军事活动中的规律。作战仿真模

型是以作战活动为研究对象,由于战争本身是一个非常复杂的实践活动,在其发展变化的过程中,受到各种外在因素和内部因素的影响,战争的发展方向和最终的结果,有其自身规律作用的必然性。对于战争过程中的具体军事作战行动,受到的影响是方方面面的,其中主要受到人为因素、武器装备因素及战场环境因素三个方面的影响。作战仿真模型与一般的模型有一定的差异,主要表现在:一是作战系统本质上是一个非线性的复杂系统,输入输出之间的关系不一定是一种固定的线性表示,且这种表示会出现经常的变化,输入输出之间不是一一对应的关系,仿真结果具有不确定性,因此对仿真模型构建需要采用相应的建模方法。二是作战仿真模型通常在一种对抗的环境中存在并发挥作用。既有友邻之间的协同与配合,又要考虑双方的对抗关系、制约因素等。三是内容描述与关系表示上更为复杂。作战仿真模型要能够反映战争对象的特性和表征,展现战争或战斗的内在规律和外在联系。作战仿真模型以军事作战行动为主要描述对象,其描述的内容和关系相比于一般的建模而言会更加复杂。

2.1.3 仿真模型体系

仿真模型体系是仿真系统的核心和基础,对于模型开发者而言,模型体系能够为其提供一个针对仿真应用的全局视图,从整体上把握系统中模型的构成及相互之间的复杂关系,对于模型的科学管理和维护具有很好的指导作用。

模型体系构建是一个较为复杂的过程,需要充分了解作战实体的结构、功能、关系、行为和交互过程等。模型是按照一定的规范对现实世界实体、事件、关系和过程的具体抽象和描述。作战仿真模型是对作战行动过程中的实体、事件、行动、交互关系和作战过程的一种抽象和具体描述^[27]。作战仿真模型体系(简称模型体系)是由作战仿真模型及模型之间关系构成的集合。研究内容主要包括模型体系设计、模型结构设计、模型描述方式、组合方式、模型之间的关系表示及模型集成框架等。

建立作战仿真模型体系的作用表现在以下几个方面:①在统一的模型框架和体系下,从模型描述、模型设计到模型运行均采用相同的方法,增强模型的统一性;②模型体系能对所有模型进行系统分类和管理,建立模型之间的相互关系,便于建模仿真人员清晰地理解模型之间的交互关系、指挥控制关系、隶属关系及

组合关系,增强模型的应用性;③通过统一的方法描述模型的粒度、接口定义、模型约束,确保模型语义和语法的一致性;④模型体系建立后,可以按照某种方法对模型之间的多种关系进行定量和定性分析,进一步抽象和发现模型之间的本质关系,把握作战实体、作战过程和作战行动的规律性。

作战仿真模型体系需要从作战仿真应用的需求出发,对参战力量、主要行动和作战空间进行整体规划。这个体系,纵向应覆盖战役层和战术层等多个层次,即“纵向到顶”;横向应包括陆军部队诸兵种(专业),即“横向到边”。从内容上讲,应该包括装备应用、作战行动的执行过程、各级指挥机构的指挥行为及作战行动对战场环境产生的影响。

2.2 仿真模型构建要求

2.2.1 可信性

随着战争形态的不断演进,特别是信息化战争对精确作战提出了更高要求,基于信息系统的体系作战使得作战过程中的任何一个环节都十分重要,任何一个环节的失败都有可能导致战争的惨败。通过仿真手段对战争进行预实践和推演是研究现代化战争的一种重要方式,因此对作战实体、作战行动、打击效能的仿真的精确化程度关系到整个推演结果,从而对仿真模型提出了更高的标准,要求不同类型的仿真模型在一定范围内必须是可靠的、可信的。可信性具体表现为以下几个方面:一是模型应能基本反映仿真对象的基本特点,特别是信息作战中的行动特点,在表现形式上具有可信性,让参与者能够感受到一定的实境感。二是仿真模型的输入/输出要能够基本依据作战过程中的现实情况,除不可避免的随机因素外,仿真的结果要有可信性,如作战实体的行动速度应根据不同的战场环境相应变化,在火力毁伤仿真计算中,应充分反映战场地形、幅员、队形、目标性质等因素,以体现作战实际。三是模型规则是透明的,而不是“黑盒”,规则数据不是一成不变的,而是可以根据需要随时进行调整的,能通过灵活性的设置增强模型的可信性。

2.2.2 重用性

作战仿真建模需求是不断变化的,模型往往是针对特定任务而开发的,一旦换一种应用场景,模型就要重新开发。一方面模型资源不能得到很好的再利用,不同系统之间的模型兼容问题日益严重,使得系统之间互联更加困难;另一方面,模型构建是作战仿真的核心,花费了大量的人力资源成本,不能重用的模型会严重降低建模的效益。实现模型的重用需要以下几个条件:一是要建立一套可以遵循的标准规范,使得对建模流程、建模元素和建模方法等进行标准化。二是要能够提供一种通用的建模框架,使得模型的构建能够以相同的方式进行描述和管理,便于模型间的互访问。三是要以模块化、组件化的方式构建模型,每一个可重用的模型必须以一个可分离的方式独立存在,通过标准的输入输出接口定义实现模块功能,同时模型也应该是参数化的,模型在实例化过程中,通过加载不同的参数生成不同的模型。四是模型结构的层次化。通过层次化的组织形式,使得模型能够以一种更灵活的方式对复杂模型的构成进行重新组织,在层次化的布局中,通过一定的组合规则,形成若干新的仿真模型,从而实现模型在更大范围内的重用。

2.2.3 可扩展性

可扩展性实际上是指在建模过程中的一种适应变化的能力。从建模方法上看,这是一种设计方法上的可扩展。对于模型的可扩展性要求主要是基于以下考虑:一是仿真业务的需求具有不断增长性和逐渐丰富性,随着作战仿真应用的不断深入,对模型的构建也会提出更多要求,可扩展性就是顺应需求发展的内在表现。二是丰富模型体系的客观要求,体系是相互联系的系统构成的有特定功能的有机整体,体系也处于一个不断的变化过程中,作战仿真体系随着作战要素的增加而不断丰富,因此体系的丰富需要模型的扩展得以支撑。

可扩展性表现为以下几个方面:①模型功能上的可扩展。一个模型构建完成后,并不是一种终结状态,而是可以在当前的状态下,通过扩展插件或者增加接口的方式形成新的更多的功能,便于模型快速重构。②模型体系规模上的可扩展。

模型体系扩展需要在统一模型框架下进行继承扩展,一个模型可以通过继承或者多态的方式快速重构多个其他功能的模型,扩展性是实现模型体系构建的一种重要基础。③建模流程的可扩展。框架虽然规定了建模的一般过程和步骤,但是可以通过增加活动节点和控制逻辑,实现建模过程的扩展性,这样建模在统一框架下,每个模型的复杂度不一样,即便是不同分辨率的模型,也能够实现有效互访问。

2.2.4 易维护性

易维护性体现的是模型构建与使用过程中的可修复和改进的难易程度。易维护性是实现模型可信、可靠的重要基础,有的模型需要经过多次的维护改造才能真正达到可信、可靠。易维护性主要表现在以下几个方面:一是模型编辑上的易维护性。也就是说随着模型功能的完善,模型应该能够便于修改,只要通过简单的操作就能实现功能的更新和拓展。二是模型管理上的易维护性。随着模型数量的不断增加,不管是概念模型、逻辑模型还是程序模型,有序管理对提升模型的使用效率至关重要。特别是对于程序模型,在管理维护上应能够提供版本管理、分类管理及查询管理等功能。三是模型存储上的易维护性。当前,对于模型的管理,各个用户只是根据自身的需求组织,缺乏统一的存储,更多的模型是直接“存放”于应用软件系统之中,没有能够按照统一的方式对模型进行标准化存储,要实现存储上的易维护,需要规定好存储的格式、方式、位置、分类和名称等。四是模型使用上的易维护性。主要是模型以何种方式向外提供服务以实现模型共享,是按需服务还是全部共享,是授权使用还是直接调用,在建模之初应该予以明确。

2.3 作战仿真模型体系

随着作战仿真应用规模不断扩大,专业化、精确化、体系化程度不断提升,使得仿真模型体系建设的重要性日益突出。一方面,由于模型开发的技术体系不同,使得开发的模型通用性差,很难在一个体系框架下进行重用,在构建新的仿真系统时往往需要重新开发,极大地浪费了人力、物力、资金和时间;另一方面,

随着军队改革的不断推进,新的编制体制调整后,武器装备性能、作战行动样式发生了较大变化,以往的模型体系难以支撑新的仿真应用。

2.3.1 要素组成

仿真模型体系是对模型构成及其相关关系的系统描述。模型体系是对模型分析、层次结构及其相互关系的抽象化表达。现代战争的信息化程度提升,提高了对作战仿真精确化的要求。信息化战争中,各兵种及保障部队的行动专业性强、复杂度高、协同关系多、影响因素多,系统地研究作战仿真模型的构成、分类结构及关系描述是构建科学、合理的仿真模型体系的关键。这是一项十分重要的、基础性的顶层设计工作,模型体系的构建是进行模型开发与管理的核心,模型体系可以抽象出作战域范围内的特征实体,以及各个实体之间应有的复杂关系。对于仿真模型体系的描述,主要应包括模型体系架构、模型分类框架、模型层次结构和模型关系矩阵视图。

体系架构是一种对体系及其要素之间关系的描述模型,其目的在于对体系结构的表现形式、组织关系及呈现方式等进行综合描述,在构建模型体系过程中,需要按照上述逻辑关系对模型进行组织和管理。模型体系架构如图 2-1 所示。

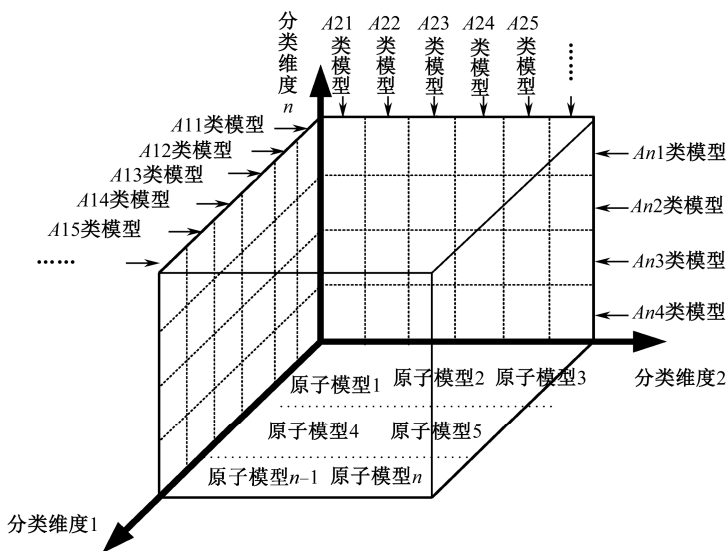


图 2-1 模型体系架构

模型体系通常由各种模型组成，必须采用一种合适的分类方法，使模型在归类过程中具有唯一性，因此需要建立一种模型分类框架。这种框架的作用就是，不管什么样的模型，不管按照什么分类方法，使其总能落到某一类框架中，并且一旦属于某一类，总能有与之相应的分类标识，如图 2-2 所示。

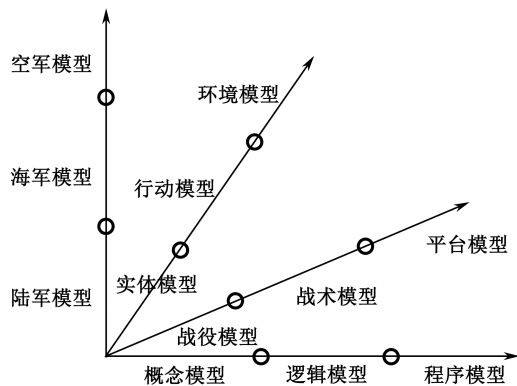


图 2-2 模型分类框架

模型层次结构是采用某一维度对模型体系进行有序划分的描述形式，根据需要可以建立多层次的模型结构，A1,B1,C1 属于第一层次中的不同类别模型，A1,B1,C1 分别可以逐级往下分解，直到不能再分为止，具体如图 2-3 所示。

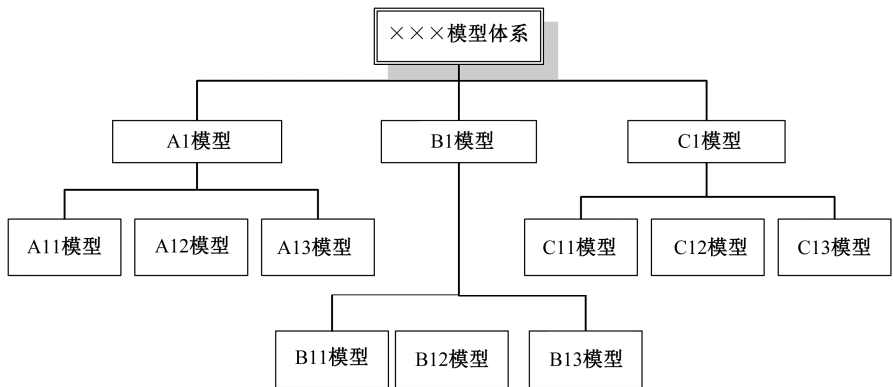


图 2-3 模型层次结构

模型关系矩阵视图是模型相互之间关系的描述，模型与模型之间可以有一到多种关系，在构建模型体系时，这些关系应予以明确，具体如表 2-1 所示。详细的关系描述可以进一步细化。

表 2-1 模型关系矩阵视图

	A 类模型	B 类模型	C 类模型	D 类模型	E 类模型	F 类模型	G 类模型	H 类模型
A 类模型	■	■						★
B 类模型		★				■		
C 类模型	■	■		●	■			
D 类模型		■	■			★	■	★
E 类模型			●		■		■	■
F 类模型						■		
G 类模型	■	★	■	●	■	■	●	■
H 类模型	■	■	●	■				

注：■、★、●分别表示相互之间有映射、交互和控制关系。

2.3.2 体系内容

陆军作战过程复杂，面临的战场环境多变，涉及的作战样式、作战行动多，覆盖的兵种专业多，协同关系复杂，因此对其作战过程进行仿真需要完备、可信、配套的仿真模型体系，以支持陆军作战训练、装备发展论证、作战概念验证和战法研究等。

从作用域来看，建模仿真需要构建一个涵盖物理域、信息域和认知域的仿真模型体系。实际上，这三个域是从不同视角、不同维度对战争的抽象描述，也是战争胜负的三个核心域，从这三个域对作战问题进行建模仿真才能更好地体现和反映战争的本质。物理域主要是指各类作战实体、各种武器装备平台实体模型等，信息域模型主要是指电子对抗模型、网络通信模型、传感侦察模型、电磁环境模型和信息传输模型等，认知域模型主要是指行动智能体模型、指挥决策模型、自主交战模型等。

从层次来看，建模仿真需要构建一个贯穿联合战役、军种战役、合同战术、分队战术和单武器平台等多个层次的模型结构体系。对作战问题的建模仿真必须是以体系的方式呈现，特别是信息化战争对体系对抗的要求尤其突出，随着信息

技术的发展及智能化技术的快速跟进,任何一场战争或一次战斗将不再是单打独斗,都需要体系力量作为支撑,因此对战争和作战问题的研究必须是成体系的,需要构建贯穿联合战役、军种战役、合同战术、分队战术和单武器平台的纵向仿真模型体系,以实现对联合作战背景下陆军作战仿真的多种需求。

从类别上看,建模仿真需要构建一个包括战场环境、武器装备、作战行动、指挥控制、分析评估和综合作用等多种类型的模型体系。对战场环境的仿真是整个建模仿真的基础,是影响作战进程的重要因素,主要包括地理环境、气象水文环境、电磁环境等方面的仿真。这一类仿真除了对其现象进行客观描述外,更多的是该类模型要能与作战行动模型进行交互,以体现战场环境对实体行动的影响。对武器装备的仿真是整个建模仿真的关键,是战斗力指数的重要体现,主要包括各类武器装备的战技术性能、作战运用效能等方面的仿真。这一类仿真重点是对装备物理属性和操作过程的仿真,反映的是具有完成任务的战斗能力。对作战行动的仿真是整个建模仿真的核心,是构建战争交战关系、行动链路的重要方式,主要是对行动实体与指挥实体之间、行动实体与其他作战行动实体之间发生的一系列过程和活动等的仿真。这一类仿真重点是对各方交战行动及其结果的仿真。另外,对指挥控制的仿真体现的是指挥决策与行动控制的内容和规则,是驱动仿真不断向前推进的重要途径,主要是完成对实体指挥控制活动及其外部形态、状态的仿真。这一类仿真重点是指挥技术与战术运用的仿真。

从应用对象上看,建模仿真需要构建一个涵盖合成、炮兵、防空、陆航、特战、作战支援、勤务保障等多兵种专业的模型体系。随着陆军部队编制体制的调整,各种专业也进行相应的整合,为了满足新形势下陆军作战仿真的需求,必须要能够按照现有兵种专业的划分构建相应的仿真模型体系,以满足不同作战编组下的作战仿真。合成兵种的仿真主要是完成装甲步兵、装甲突击、火力和支援保障力量及其行动的仿真。

2.3.3 设计方法

关于模型体系的设计方法很多,根据应用需求的不同,可采用不同的设计方法。这些方法包括框架式方法、元模型方法、面向对象建模方法及组件化方法等。

1. 框架式方法

框架式方法主要是以模块化方法定义模型,为系统建模提供一种有效的建模仿真框架,通过采用统一的标准和规范建立模型并构建模型与模型之间的关系,采用通用的模型表示和描述方法,使模型在更高层次具备可重用、可组合和互操作能力。

2. 元模型方法

元模型是模型的模型,是对如何建立模型、模型的语义,以及模型之间如何集成和互操作等信息的描述。元模型方法主要是完成对某一特定领域建模环境的规范、交互关系、语法和语义定义,通过元模型方法可以获得比模型的抽象程度更高、集成度更高的模型。在元模型的设计上,充分考虑各种已有模型的集成需求,使其成为已有建模技术、建模工具和建模语言等的集成基础,实现对不同层次的抽象建模。元模型具有描述建模语言的语法和语义的能力。

3. 面向对象建模方法

面向对象的建模方法通常以类为基础,类具有通用的属性和方法。面向对象的建模实际上就是完成抽象、封装、继承和多态的过程。通过抽象去除模型的无关细节,降低模型的复杂度;通过封装将模型对象的内部实现从公共接口分隔出来,便于其他模型与其交互;通过继承减少相似模型的属性和方法重复建模,可以直接引用或自行定义,增加模型的层次性。此外,对象间的泛化、特化、聚合、关联及多态关系,可以用于对仿真系统中各种对象及对象之间的关系建模,进而建立仿真模型体系。

4. 组件化方法

组件化方法的核心是基于接口的程序设计,每个组件都会提供一个定义好的接口,可以与组件进行交互。组件化方法的最大特点是复用性,使模型具有继承性,通过对模型功能 and 数据流程的抽象和分析,抽取其中的复用部分,能够在更高层次上提供组件组装的基本框架,解决模型组装、组合和重用问题。

上述方法有各自的优缺点,在建立模型体系时,不能局限于单一的建模方法,而是需要综合多种方法,充分发挥每种方法的优势,因此,在进行模型体系设计

时,综合使用上述方法,使其既能充分利用继承、多态和聚合等方法对系统建模,又能利用推理的方法对模型体系进行分析。

2.4 作战仿真模型体系应用

2.4.1 模型共享

模型作为一种基础资源,经过多年的积累,各军兵种已经积累了相当数量的仿真模型。由于在设计、实现和使用方式上的差异,加上对模型的知识保护,模型共享问题一直未得到很好的解决。一方面,现有的模型成果有其固有的应用模式或者嵌入在某一专用系统中使用,由于建设标准的问题始终在领域难以得到统一和规范,尽管需求迫切,但操作实现困难;另一方面,现有的模型资源具有自主知识产权的专用资源和安全的特有资源,管理机制的不顺畅导致模型共享难,同时,不同仿真需求对聚合与解聚提出了更高的要求,但不同建模方式构建的模型很难实现有效的聚合与解聚。不管是概念模型、逻辑模型还是程序模型,都可以以服务的形式向其他用户提供共享。但是从应用层面看,对于模型的直接共享应用更多的是针对程序模型而言,实现对不同类型的程序模型需要制定一套标准规范,采用一种封装和服务化的思路,设计统一共享访问接口是进行模型充分共享的有效途径。

实现模型共享的基本条件:一是要构建统一的模型体系,不管什么类型、什么粒度的模型都能在模型体系中找到定位,便于模型的规范管理与使用。二是制定建模标准,使得模型开发按照统一的规范进行,便于模型在更大程度上融合。三是构建模型共享平台。模型版权者向模型库提交模型及服务目录,使用者共享模型服务,无权也不能占用或修改模型资源,各用户从模型资源库中提取的是模型服务及其相关数据。四是建立模型共享机制。在建模仿真领域,全军要建立统一的领导机构,建立模型共享、共用机制,在制度上为模型共享提供支持。

2.4.2 模型服务

模型服务是一个平台独立的、低耦合的、自包含的、基于可编程的应用程序，以模型服务的方式向客户提供模型资源是实现模型共享的一种重要方式，可以通过开发的标准来描述、发布、发现、协调和配置这些服务。仿真模型服务是指基于计算机网络技术，将分布在不同地域、用于实现特定军事仿真模型计算或存储的节点，以网络互联的形式构建仿真模型资源服务体系，为用户提供能够共享与可重用的仿真模型资源的活动过程。构成模型服务的基本要素主要有服务主体、服务客体、服务资源和服务管理。其中，服务主体，是提供服务的一方；服务客体，是申请或接受服务的一方；服务资源，是指能够实现服务功能的知识、工具、数据等各种资源。通常，服务主体与服务客体是多对多的关系，一个服务主体可以服务于多个客体，一个服务客体也可以请求多个服务主体为之提供服务。在以服务为导向的架构中，对用户而言，所谓“服务”，是指独立于技术实现的并具有相对独立功能的程序组件，以一种统一的规范对模型描述进行封装后，按照统一的接口就能实现服务的访问。采用面向服务的设计思想与方法构建模型共享系统，在适应需求变化和功能进化方面具有更高的灵活性，在资源的共享与重用方面具有更广阔的空间。这种应用模式使得技术与应用相分离，使用者无须关心模型本身的具体细节，只需要通过一定的配置，模型就可以在更高层次上进行灵活应用。

服务方式是一个相对抽象的概念，从不同角度可以得出不同的分类。从服务的距离空间来看，可以分为远程模型服务和本地模型服务；从服务的具体内容来看，可以分为概念模型服务、逻辑模型服务和程序模型服务；从服务推送的形式来看，可以分为按需服务和自主推送服务；从服务的提供方式看，可分为模型下载服务和在线服务；从服务的寿命周期管理看，模型服务包括注册服务、存储服务、更新服务、管理服务和应用服务等。通过模型服务的方式，能够实现仿真模型资源在更高层次的共享与重用。一方面，模型以分布式方式在统一共享平台下，按照仿真模型全寿命周期管理的要求，模型资源建设能力会不断增加，模型服务能力也会逐渐增加，能够极大地提高模型的管理和使用效益；另一方面，在统一共享平台支撑下，各用户单位无须花费太多的时间精力对模型本身进行维护。使

用者不需要关心模型的内部结构。对于模型的有效性,由专门的权威机构对其验证,使得模型能够在一种更加开发的形态下进行应用和拓展。

2.4.3 组合应用

组合应用的目的在于快速构建新模型,拓展模型新功能,提高模型重用性。模型组合是一个规范化、体系化的过程。一个复杂模型功能可以通过几个简单的模型组合而成。简单模型具有独立的应用功能,通过各种不同粒度的基本功能模型构造更高层次的复杂模型是进行模型组合的主要任务。模型组合应用的内在驱动在于不断增长的需求,应用在不断拓展,需求在不断变化。满足需求的最好方式不是一味地增加模型数量,而是需要建立一种机制和方法,通过对现有模型的组合使其具有更多的模型功能。

在仿真模型体系的统一支撑下,应用模型组合是最大限度发挥模型效益的重要途径。一次作战行动通常由若干个作战活动组合而成,对于此类行动仿真需要多个模型共同作用才能完成。现有模型服务的开发大多都只是细粒度的单个行动仿真或简单分析计算仿真。当用户的模型共享需求为能实现某战役作战任务或复杂功能的模型服务时,依靠单个模型服务是不能满足需求的,需要将相关的多个行动或计算分析模型服务进行组合应用。模型组合应用包含以下两个层次:一是建模层次的组合。组合建模是一种灵活建模的方式,在对模型进行设计时,各类模型均以最小单元的组件形式存在,每个组件既可以作为独立的功能体,也可以将多个组件按照一定的方式组合后形成功能综合体。二是应用层次的组合。应用层的模型组合主要是通过服务组合的方式实现,各应用功能不管粒度如何,按照服务的方式进行描述、注册和封装,根据申请者对应用功能的需求及需求的复杂程度,若单一功能需求,则通过直接调用的方式完成;若功能需求相对复杂,则首先通过自动解析用户需求。而后按照一定的配置规则,将相应功能服务经过组合后提供给用户。用户体验到的只是模型服务功能,至于采用何种方式获得的功能,用户是无法感觉到差异的。

2.5 本章小结

本章通过对作战仿真建模相关概念的介绍，阐述了仿真模型构建要求，同时对作战仿真模型体系的要素组成、体系内容和设计方法等进行了分析，最后从模型共享、服务及组合应用等不同视角对模型体系应用问题进行了探讨。

第 3 章

作战仿真模型体系分析与建模



战争的复杂性决定了作战仿真模型体系是一个复杂系统,作战仿真模型体系设计就是要以模型重用与互操作为目标对复杂系统进行分析,提供一种有效的方法用于指导作战仿真模型的设计与开发,以满足作战仿真系统对模型的要求。仿真模型体系对于模型的开发具有一定的指导作用,模型体系主要是从模型构成的角度考虑模型组成及模型之间的相互关系。

3.1 仿真模型军事需求分析

本书以陆军部队为主要研究对象,其目的是为在更大范围、更高层次设计仿真模型、构建模型体系提供通用的方法与手段。因此,本书所涉及的模型及模型体系,不仅能用于陆军部队的作战实验、装备论证、仿真训练等,更能对联合作战仿真提供有效的模型支撑。

3.1.1 模型特点分析

本书主要是对一种灵活、通用的模型及体系的一些关键问题进行研究,通过对陆军部队作战编组、作战行动和特点的分析可知,仿真模型应具备以下特点。

(1) 分布式交互。当前陆军部队使用武器装备和信息系统呈现分布式部署。在对其作战行动进行仿真过程中,需要各种相应的实体模型能够进行交互,并能在分布式环境下进行仿真应用。

(2) 统一共享。这是陆军部队各作战单元与作战要素信息共享的具体体现,通过专门的信息组件或信息处理单元完成对战场各类信息的采集、传输、处理和发布等,使各种作战实体能够按照相应的信息共享方式获取战场各类信息,使模型具有统一共享的能力。

(3) 多分辨率生成。仿真模型是对军事领域的抽象,对于陆军部队而言,既要满足合同战术(合成师、旅、团)、分队战术(营、连及主战信息化武器平台),也要满足军种战役仿真的需要。针对不同的仿真目的,需要建立不同层次的仿真模型,因此,陆军作战仿真模型应能支持多分辨率生成。

(4) 灵活重组。陆军部队作战实体类型多、层次多,如果针对每一种类型、每一个层次都建立相应的仿真实体模型,难度比较大,这就要求在建模时提供灵活重组的方法。通过一些基础模型或组件的重组实现不同的仿真实体模型,满足按需建模的要求。

(5) 模块化组装。为了突出陆军部队作战行动的特点,模型必须是组件化、模块化的,便于组装成新的模型。信息化装备、系统及其侦察方式等通过组件的功能实现,组件的组装能够实现基于信息系统的体系化作战仿真能力。

(6) 自动组合。在一定的仿真约束下,通过对用户功能需求的分析,模型能够按照相应的接口规则、组装规则和消息传递规则等自行组合,具备语义理解和匹配的推理功能。

(7) 透明封装。由于模型体系是一个相对复杂的系统,其中包含各种类型的模型,最终用户不太关心模型具体是如何实现的,而是关心模型能够提供何种功能、如何提供功能,因此,需要按照统一的方式对模型进行封装,使其按照统一的接口对外提供仿真功能。

(8) 多态继承。多态是可以对相似的接口使用不同的实现方法,而继承是一个类具有另一个类的属性和方法的能力,多态继承方法的结合使用可以提高建模的灵活性和层次性。

(9) 框架式设计。框架是实现模型一致性的基础,按照统一的建模框架可以避免模型访问过程中的歧义性,在统一的模型框架和体系下,从模型描述、模型设计到模型运行均采用相同的方法,能够增强模型的一致性。

(10) 聚合与解聚。陆军作战仿真涉及多个不同层次的模型,通过模型间的聚合与解聚实现模型在不同层次间的映射和转换,为多分辨率建模提供一种有效的途径。

3.1.2 模型目标分析

单独使用面向实体的直接建模方法难以满足作战仿真的需要,因此,需要改变以往直接面向实体建模的方式:首先构建各种不同类型的模型组件,而后通过组件装配与组合的方式实现对各种实体的建模。在此基础上,通过聚合与解聚的方式实现对不同作战层次模型的建模。建立陆军仿真模型需要实现以下目标:

(1) 可重用性。重用既是对模型的基本要求,又是模型体系的一个重要目标,也是设计模型体系的一个基本原则,通过重用减少重复建模。

(2) 平台无关性。模型应能进行独立封装和发布,通过提供定义良好的显式接口对外提供模型功能,屏蔽模型的内部实现细节和数据结构,使模型的应用不受平台和环境的约束。

(3) 可组合性。重用是组合的前提,组合是重用的一种具体形式,通过对基础模型的组合实现对模型的扩展,以满足多层次、多类型仿真模型的需要。

(4) 领域表示一致性和可理解性。通过类模板和继承定义实例完成,使模型在统一的框架下设计,规范模型描述,减少模型的歧义性。

(5) 可扩展性。向外部扩展更多的对象和层次,向内部扩展更高的模型解析程度。

(6) 协同性。模型之间能够进行交互,并按照一定的规则完成作战任务的协同。

在对陆军部队装备进行建模时,要体现其数字化、网络化、信息化和一体化等特征;针对陆军部队作战行动特点,要体现对指挥控制、情报侦察、预警探测、通信、电子对抗、定位、识别等过程的建模;针对新型陆军部队编成、编组等特点,要体现对编成、编组的灵活建模。围绕上述需求,在建模过程中主要解决以下几个问题:一是组件化建模问题;二是元模型建模问题;三是模型体系构建问题;四是组合建模问题。

3.1.3 组件化建模需求分析

陆军部队的作战样式已经由传统的机械化作战转变为基于信息系统的体系作战，因此，传统的仿真模型已经不能适应新的作战样式，需要开发新的模型；同时，体系作战的仿真建模比传统仿真建模更为复杂，传统的面向对象的软件开发方法对于解决复杂系统具有很大的难度，不易实现，需要新的建模方法来解决这一难题。

面向陆军部队的仿真软件系统由于具有很高的复杂性，需要通过一种能够将复杂军事问题分而治之、降低系统开发复杂性的方法，来进行复杂仿真系统的开发。基于组件的开发方法的基本思想就是“分治”，强调将复杂模型分解成松散耦合的组件独立开发，然后通过接口和脚本将独立组件连接起来，组合成更为复杂的模型。基于组件的开发方法与我军目前仿真模型体系开发方法的需求不谋而合，因此，组件化建模应成为军事建模的主要建模方法和手段。

如今，模型的开发难度越来越大，研发成本越来越高，开发周期越来越长，而随着陆军部队建设步伐的加快，部队作战样式日新月异，作战方法不断创新，仿真模型体系开发速度远远不能跟上军事领域的高速变化和快速发展。目前，如何解决复杂军事仿真模型标准和开发效率问题已经成为当务之急。

当前，军事仿真模型系统基本上都是采用面向对象的方法开发，虽然面向对象的方法，在代码功能的封装性和安全性上有较高的保障，但是，其模型复用的层次较低，只能进行代码级复用。代码级复用存在两个主要弊端：一是使用已有代码会牵扯到对复用代码的改动，造成代码冗余；二是系统开发中会因为代码不熟，对复用代码修改不全面等问题，给模型系统带来质量问题。

基于组件的方法有两个显著优势：一是由于组件之间采用预先定义的接口进行相互连接与调用，使得组件呈现了较高的复用性，并且复用层次较高，能够提高模型开发效率、降低模型开发成本；二是组件化开发通过逐步分治，能够将模型分成多个独立、松耦合的组件体，便于进行模型组合、聚合与解聚，也能够极大提高模型的使用效率。

3.1.4 组件化建模作用分析

模型构建方法既要具有一定的通用性,又要满足陆军部队的专业性,并反映出陆军作战仿真的特点,组件化建模方法能够很好地处理和解决这一矛盾。采用组件化方法构建模型体系具有以下作用:

一是通过组件提供基本的建模方法。组件可以是平台组件、传感器组件、武器组件、通信组件和辅助决策组件等,这些组件可以满足对作战实体的建模仿真。

二是通过组件体现陆军部队仿真模型特点。陆军部队的装备特征、指挥体系特征及作战行动的特征,均可以通过开发相应的组件来体现,因此,对陆军部队的描述可以简单地认为转化成了组件的描述,标准化程度增强。

三是通过组件不断丰富完善模型体系。组件是模型体系的基础,每一次模型的增加,可能只是某个组件的增加,而不需要对整个实体重新建模,部分组件可以实现重用。

四是通过组件的组合实现对不同作战实体模型的建模。作战实体通常有多个组件共同装配而成,对于复杂的行动实体或指挥实体,可能是在此基础上的组合,从而形成更为复杂的模型,避免了对复杂模型的重新开发,提高资源利用的效率。

五是通过组件的实例化,实现对不同类型的仿真实体建模。同一个组件可能由于参数的不同会产生新的实体。以飞机平台组件为例,如果加载的发动机的属性是战斗机发动机的参数,则飞机平台就是战斗机平台。如果加载的发动机的属性是运输机发动机的参数,则飞机平台就是运输及平台。其他组件的实例化以此类推。这样就只需对某一种装备实体建模,而不是对每一个装备实体建模,极大地提高了建模效率。

组件化建模可以解决陆军作战仿真模型构建问题。对于与一般的相似或相同的系统、装备,可以采用通用组件解决;对于升级改造的装备,可以在已有的平台和系统基础上,通过增加相应的组件,实现新装备的功能,以反映陆军部队装备的特点;对于全新的装备,可以通过定制开发平台组件和功能组件,而不需要重新建立一套模型体系。

3.2 模型体系构建分析

对模型体系的分析实际上就是完成对军事概念空间实体、行动、关系和过程的抽象分析,通过抽取共同的属性,按照一定的原则,将不同类型、不同层次、不同分辨率的仿真模型进行有序划分,使其在纵向上相互独立、不重叠,在横向上彼此关联、有交互。

3.2.1 模型体系构建要求及原则

1. 建模要求

军事建模与仿真是研究军事问题、探索军事活动规律的基本途径和手段。自 20 世纪中期始,美军长期从事作战仿真的军事人员,利用不同层次的多种作战模拟系统,对战前作战方案和计划论证进行仿真推演,取得了显著成效。同时,军事仿真技术在战略分析与规划、指挥员指挥训练和武器装备建设等方面得到了广泛应用,军事建模与仿真技术被美国国防部列为十大关键技术之一^[28]。我军从 20 世纪 80 年代开始,在院校、部队和科研单位展开了训练模拟系统的研制和模拟训练方法的运用,建立了一批工程标准和技术规范,开发了一批适应不同层次训练需要的模拟系统,取得了明显的效果。我军的军事建模与国外发达国家相比还有较大差距,应从以下几个方面进行改进和完善:

(1) 提高模型的置信度。随着信息化战争形态的不断演进,基于信息系统的作战体系较传统作战体系的复杂程度成级数增加,运用现代军事运筹学理论和建模方法来认识并描述其模型比较困难。现有基于结构级、行为级和基于 Agent 的战争系统建模方法构建的作战模型,还无法准确反映信息化条件下作战的基本特点,且通过系统分析而产生的数学模型可靠性也比较低。另外,系统从军事概念模型到数学模型,再到程序模型的系列转换中,需要仿真开发人员进行人工理解和转换。然而,由于技术人员与军事人员知识背景的差异,因此,往往在进行模型转换过程中造成模型“失真”。

(2) 提高模型重用性。由于以往的模型开发缺少统一的技术框架支持,大部分都是依托具体平台实现的,当平台需要变更时,原有模型可重用性较弱。长期工程实践表明,对于大型复杂软件系统开发,必须首先构建系统模型体系,以改善软件系统开发的效率并提高模型的可重用性^[29]。提高模型重用性的一个基本前提就是采用统一的模型框架,为模型描述、设计和访问提供标准的方式。

(3) 提高模型可维护性。在美军 CMMS 的描述方法中,早期使用的专用方法针对具体情况进行。随着作战规则的变化,需要对军事概念模型进行重新修改、维护时,往往无法获取模型的充分信息。这种模型重构的方法,只能一边分析仿真代码,一边研究模型文档,并向军事人员咨询,降低了模型的开发效率,且使得大型系统的模型维护变得几乎无法完成。在我军以往的仿真系统当中,模型实现基本上也是通过将业务逻辑与具体的平台或者编程语言紧耦合实现的。因此,当模型需要进行改进时,就必须从概念层到数学模型层、再到代码层对原有模型进行修改,这样大大增加了模型维护的成本。同时,陆军部队作战行动建模是一个不断探索的过程,需要在作战模拟的战争预实践过程中不断修改和完善,这就对作战仿真模型的更新和维护的时效性提出了很高要求。

2. 建模原则

1) 重用性原则

注重模型结构重用和使用方式重用。模型重用的主导思想是避免重复建模,尽可能降低仿真开发成本,模型重用是构建大型、复杂模型的基础。按照仿真活动过程,可分为概念模型重用、数学模型重用和程序模型重用。为了实现模型重用,模型体系的各个模型需要按照统一的结构、规范表示。

2) 完备性原则

模型体系的完备性是指所建立的仿真模型完全覆盖描述的对象,能够完成面向服务的作战仿真的所有内容。

3) 统一性原则

模型可以根据不同的类型和用途进行分类,不同类型的模型,其框架结构可能不同。模型体系的设计是决定模型是否可重用、重组、扩展和维护的关键。要实现模型在军事范围内的通用、重用和组合,模型的分层和分类必须统一,即模

型体系框架具有统一性。

3.2.2 模型体系构建所面临的问题

仿真模型体系的构建是一项复杂的系统工程，需要解决和突破很多问题，如模型开发的技术问题、模型共享的体制问题、模型存储的方式问题及模型应用的模式问题等。从模型开发的技术层面看，由于仿真模型本身种类多样、关系复杂，从军事角度梳理出它们之间的逻辑也是一个比较困难的问题。具体来说，模型体系构建过程中面临的问题包括仿真模型自身的复杂性、建模过程的复杂性和模型之间关系的复杂性。

1. 仿真模型自身的复杂性

从作战仿真对象来看，主要是对战场环境、武器装备、指挥决策、交战过程及作战结果分析进行建模，逻辑复杂；从仿真模型的类型来看，主要包括机动模型、传感器模型、电子对抗模型、平台模型、通信模型、行动模型、认知模型和武器模型等，类型多样；从仿真模型的功能来看，主要是完成指挥决策、作战行动、分析评估、装备运用和军事力量仿真等。其中，对武器装备运用进行仿真是重点，还要涉及对交战过程的仿真，由于其涵盖的范围广、过程难和种类多，使仿真模型自身具有复杂性。

2. 建模过程的复杂性

仿真建模过程是对作战过程的抽象，作战过程本身就是一个十分复杂的过程，涉及人员、装备及交战关系，对复杂对象或事件的抽象必然具有复杂性。具体体现在以下几个方面：一是建模过程包含多个环节，分别需要完成军事概念建模、数学逻辑建模和程序建模，并且这三个环节具有一定的关联性和约束性，需要环环相扣，逐步展开。二是建模对象所含的因素多样性，在建模过程中，需要对模型粒度、模型的表示方法、模型的约束及模型之间接口的一致性等进行明确和规范。三是建模过程的数据复杂性，数据是进行模型运用的基础，数据对于模型仿真应用具有重要作用，包括模型的元数据、想定数据、仿真运行配置数据以及模型的 VV&A 数据等。建模过程的复杂性决定了模型体系的复杂性。

3. 模型之间关系的复杂性

对于作战仿真模型体系而言,武器模型、通信模型、评估模型、决策模型和环境模型等并不是独立存在的,彼此之间存在相互的联系,并以统一体的形式对整个仿真结果产生影响。模型关系主要表现为继承、组合、交互、隶属和指挥控制关系等,其中,继承关系指采用面向对象建模方法描述的模型之间泛化和特化关系。组合关系是模型之间的接口连接和行为组合关系。交互关系是对模型相互访问、操作及数据交互的描述。隶属关系描述模型之间的包含关系。指挥控制关系描述仿真过程中模型之间的控制与被控制关系。模型与模型之间构成关系,关系之间会产生相互影响,如隶属关系的变化会影响指控关系的变动。模型之间的复杂关系是导致模型体系复杂性的重要因素,对模型体系进行定性或定量建模,是进行模型设计与灵活应用的基础。

3.2.3 模型体系组合性分析

模型体系的组合性是指模型体系设计时对模型组合的支持能力,它对模型体系设计、模型设计及模型重用有重要影响。模型组合能力决定了模型重用的能力,为了使模型体系具备一定的组合性,应首先从结构和功能上对其进行合理划分,使得不同的模型能够在不同的层次上展现,并能根据建模仿真需要,在语法和语义层次实现模型的组合。

一般地,按照应用的层次划分,作战仿真系统可以分为战役级、任务级、战术级以及工程级仿真系统,不同层次的仿真系统对模型体系都有模型组合的需求。

下面以 JWARS 和 FLAMES 仿真系统的模型体系为例,分别从战役级与任务级两个层次分析模型体系的模型组合功能与设计方法^[30]。

JWARS 是一种战役级联合作战模型体系,它的对象模型主要是联合任务空间模型 JMSM (Joint Mission Space Model),利用多继承方法表示模型的聚合,通过模型的组合实现模型重用。最基本的对象模型由 Command、Unit、Asset、C2Information、C2Element 和 HQ 等组成,如图 3-1 所示。C2Element 表示指挥控制要素,C2Information 表示指挥控制信息,HQ 表示指挥部,Asset 表示与模型执行的任务相关的装备和人员等对象,通过组合不同的 Asset 组件,构建不同的功能模型,组件之间按照相应的指控关系传递信息。因此,JWARS 模型体系

体现了很好的组合性设计思想，不仅表现在模型分类上，还体现在模型组件组合上。

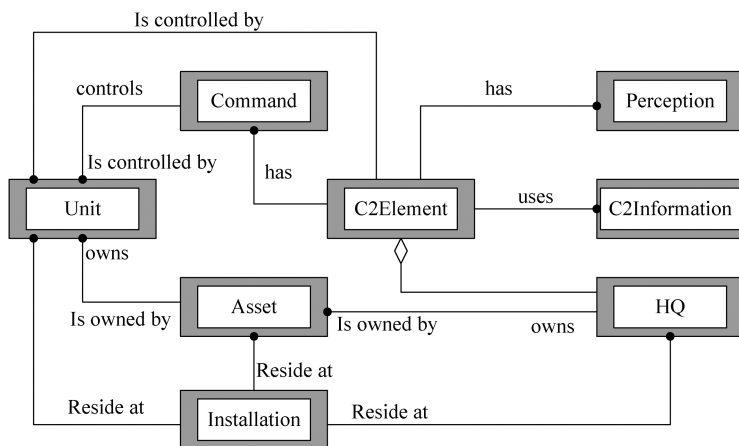


图 3-1 JWARS 的基本模型结构

FLAMES 模型体系将对象模型分为装备模型和认知模型两大类，其模型体系结构如图 3-2 所示。装备模型由平台模型、通信装备模型、传感器模型、数据处理模型、干扰装备模型、子系统模型、武器系统模型和弹药模型等八个子类构成。认知模型是对人的指挥和决策过程进行仿真，子类通过属性和方法实现模型单元功能，单元的组合通过 FLAMES 自定义的脚本语言描述。FLAMES 的单元与 JWARS 的 CompositeAsset 相似。不同的是，FLAMES 的模型组合主要通过模型设计体现，是代码层的模型组合，而 JWARS 的模型组合体现在组件层。

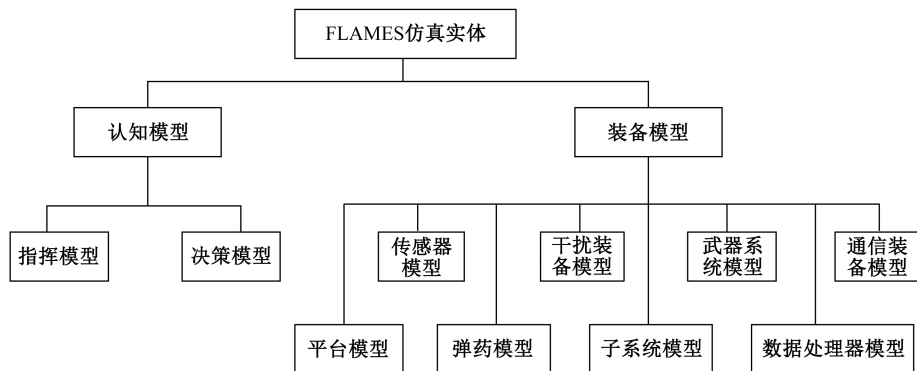


图 3-2 FLAMES 模型体系结构

图 3-3 是对 Asset 类进一步的划分。人员、供给、通信节点、传感器、武器、导航系统、油料和弹药等类都属于 Asset 的子类。CompositeAsset 子类拥有 Asset 的属性和方法及 AssetList 属性，它可以由一个或多个 CompositeAsset 或 Asset 构成。因为 CompositeAsset 是拥有运动、接受攻击和评估损耗等方法的唯一的类，因此模型一般都属于 CompositeAsset 类的实例。

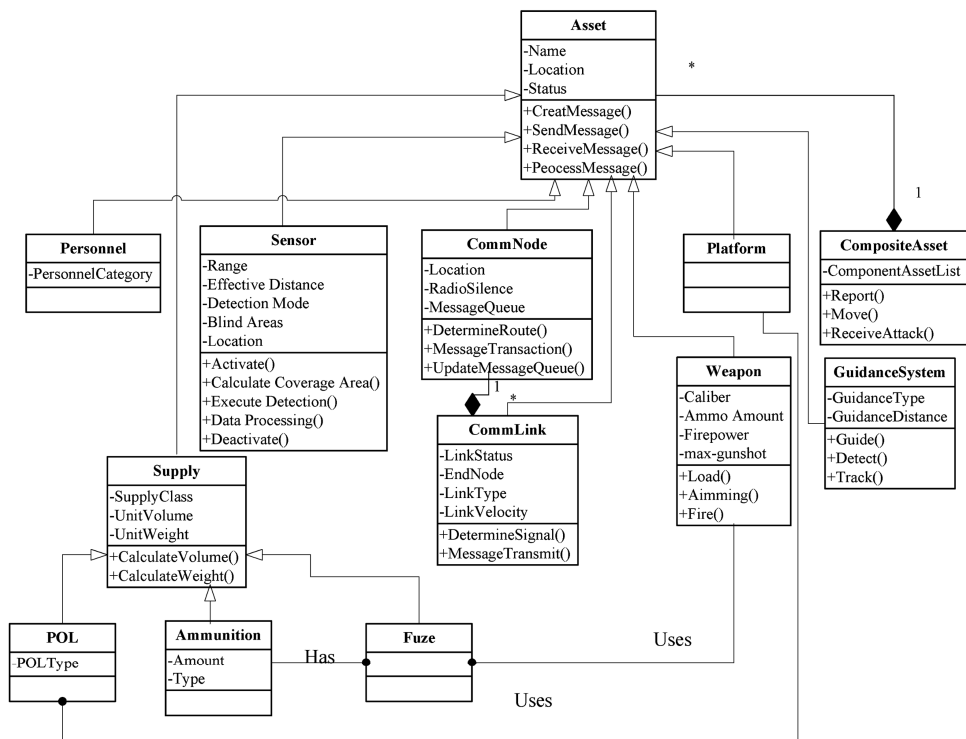


图 3-3 JWARS 的 ASSET 类和其他类之间的关系

经过分析得出如下几点结论：①上述两种模型体系的构建均以模型重用和组合为基础，且模型体系是一种易扩展的结构；②采用标准化的模型设计方法和技术框架，模型之间能够按照相应的机制、协议进行数据交互和功能交互；③模型具有统一的访问接口，便于模型间的访问和互操作；④采用面向对象的建模方法，利用多继承方法实现模型聚合，利用模型组合构建战场空间实体实现模型重用；⑤类是模型的基础，通过类间的继承、封装等实现模型的关联和组合；⑥按照一定的模型规则和方法进行组合运用，并通过消息或事件驱动的方式完成模型间的交互。上述这些特点对于指导模型体系构建具有很好的借鉴意义。

3.3 作战仿真模型体系形式化建模分析

模型体系设计的基础是结构建模,模型体系中除了包含大量的模型之外,还存在多种复杂的关系,因此,模型体系的设计首先要对其结构进行建模和分析。需要关注到模型在体系中是按照怎样的方式分类的,模型之间的关系是怎样构建的,采用何种方法对模型进行建模,使得模型在体系中具有统一性和规范性。

3.3.1 基本概念

假设某复杂系统 X 的要素集为 (s_1, s_2, \dots, s_n) , 描述系统的可达矩阵为 $M = (m_{ij})_{n \times n}$ 。

定义 3.1 没有回路的上位集 要素 s_i 没有回路的上位集记作 $A(s_i)$, 其中 $A(s_i)$ 中的要素与 s_i 无关, 而 s_i 与 $A(s_i)$ 中的要素有关, 即有向图上从 s_i 到 $A(s_i)$ 存在有向边, 而从 $A(s_i)$ 到 s_i 不存在有向边。

定义 3.2 有回路的上位集 要素 s_i 有回路的上位集记作 $B(s_i)$, 其中 $B(s_i)$ 中的要素与 s_i 有关, 而 s_i 与 $B(s_i)$ 中的要素也有关, 即有向图上从 s_i 到 $B(s_i)$ 存在有向边, 而从 $B(s_i)$ 到 s_i 也存在有向边。

定义 3.3 无关集 要素 s_i 的无关集记作 $C(s_i)$, 其中 $C(s_i)$ 中的要素与 s_i 无关, s_i 与 $C(s_i)$ 中的要素也无关, 即有向图上从 s_i 到 $C(s_i)$ 不存在有向边, 而从 $C(s_i)$ 到 s_i 也不存在有向边。

定义 3.4 下位集 要素 s_i 的下位集记作 $D(s_i)$, 其中 $D(s_i)$ 中的要素与 s_i 有关, 而 s_i 与 $D(s_i)$ 中的要素无关, 即有向图上从 s_i 到 $D(s_i)$ 不存在有向边, 而从 $D(s_i)$ 到 s_i 存在有向边。

定义 3.5 可达集合 要素 s_i 的上位集 (包含 $A(s_i)$ 与 $B(s_i)$) 又称为可达集合, 记作 $L(s_i) = \{s_j \in X \mid m_{ij} = 1\}$ 。从有向图上看, 即为从 s_i 节点出发能够到 $L(s_i)$ 节点的集合。

定义 3.6 先行集合 与可达集合相对应, 要素 s_i 的下位集称为先行集合, 记作 $F(s_i) = \{s_j \in X \mid m_{ji} = 1\}$ 。先行集合又叫前向集合, 从有向图上看, 即是所有可

到达 s_i 节点的 $F(s_i)$ 节点的集合。

要素 s_i 与其上位集、无关集和下位集之间的关系如图 3-4 所示。

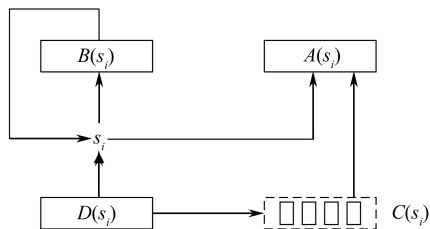


图 3-4 要素 s_i 与其上位集、无关集和下位集之间的关系

3.3.2 ISM 方法与步骤

模型体系结构的描述是一个复杂的问题，与模型的建模方法和模型体系的描述方式密切相关，需要对复杂的体系结构进行合理、抽象和简化，使得模型在整个体系中既有独立性又有关联性，方便建模人员对模型的使用和访问。本书主要采用组件化思想建立多种类型的原子模型，在此基础上，以组合的方式实现模型的重用。对于模型体系结构的建模分析主要从继承关系、组合关系、关联关系、聚合关系和交互关系等方面加以描述。

解释结构模型 ISM (Interpretive Structural Model) 是美国华费尔教授于 1973 年作为分析复杂社会经济系统有关问题的一种方法而开发的，其目的是通过将复杂的系统分解为若干个子系统，分析各个子系统之间及其内部的关系，最终将复杂系统构造成一个多级递阶的结构模型^[31]。

其基本步骤分为：①根据具体的问题，选择构成问题的要素，即系统包含的具体要素和对象有哪些。②基于某种关系构建可达矩阵。系统内的要素之间一定存在某种关系，由于建模目的不一样，这种关系可以是继承关系、组合关系、关联关系、聚合关系等。③根据可达矩阵，按照区域分解、区域内分级的思路分析并建立结构解析模型。

基于可达矩阵，将系统的要素分解为几个相互无联系，或联系极少的区域，其具体操作如下。

(1) 确定各要素的可达集合和前向集合。依据可达集合和前向集合的定义，

确定系统中各要素的可达集合和前向集合。

(2) 分析要素的共同集合。系统要素的共同集合记为 G ，其中 $G = \{s_j \in X | L(s_i) \cap F(s_i) = F(s_i)\}$ ， G 中的要素为底层要素。

(3) 区域划分。通过可达性判断建立各要素之间的关系，并以此确定这些要素的连通性，在此基础上，按照一定的规则，将系统分为有一定关联的子系统。如果要素在同一部分内，则它们的可达集的交集为非空集合，即对要素 s_i 和 s_j 而言，若 $L(s_i) \cap L(s_j) = \emptyset$ ，则它们分别属于两个连通域；若 $L(s_i) \cap L(s_j) \neq \emptyset$ ，则它们分别属于同一连通域。经过上述运算过程，可将系统 X 划分为若干个区域，记为 $\prod(X) = P_1, P_2, \dots, P_m$ ， m 为分区数。

(4) 区域内级间分解。其主要任务是将系统内的各要素按照层次进行相应的划分。根据可达集合和先行集合的定义，可得到在一个多级结构中，系统的最上级要素的可行集只能由其本身和其强连接要素组成。系统的最上级要素 s_i 必须满足下列条件： $L(s_i) \cap F(s_i) = L(s_i)$ 。具体来说，级间分解的步骤可归纳如下：①如果 $L(s_i) \cap F(s_i) = L(s_i)$ ，则 s_i 属于第一级要素；②在可达矩阵 M 中划去该要素对应的行和列，重复步骤①得到次一级的要素；③如此循环执行，直至对所有要素完成分级；④根据分级的先后次序重新对矩阵进行排列，得到重新排列后的矩阵 M_H 。

(5) 强连通块划分。为了简化问题，可以将矩阵 M_H 的行和列进一步缩减，从而得到新的递阶结构分块可达性矩阵 M'_H 。

(6) 结构模型的建立。结构模型是对元素之间关系的抽象和描述，在区域划分和级间分解的基础上，通过分析和建立结构矩阵，求解结构模型。

下面利用一个示例对仿真模型体系的结构进行分析，分别对模型的继承关系、组合关系进行建模分析。

3.3.3 继承关系建模

继承关系是面向对象设计方法的重要手段，对继承关系建模的目的在于能够对模型进行合理分类，理顺体系中模型之间的继承关系，建立模型的层次结构，利用有向图描述模型之间的关系，并形成相应的结构矩阵，使矩阵与关系一一对应，通过矩阵计算和变换使复杂的关系转换成简单、直观的结构模型，便于在程序设计阶段能够以一种最简化的方式实现模型间的组合运用。

对图 3-5 中的模型分别进行编号, 令 a_1 为仿真实体, a_2 为机动模型, a_3 为传感器模型, 一直到 a_{21} 为分析评估模型, 可以得到模型的集合 $M = \{a_1, a_2, \dots, a_{21}\}$ 。

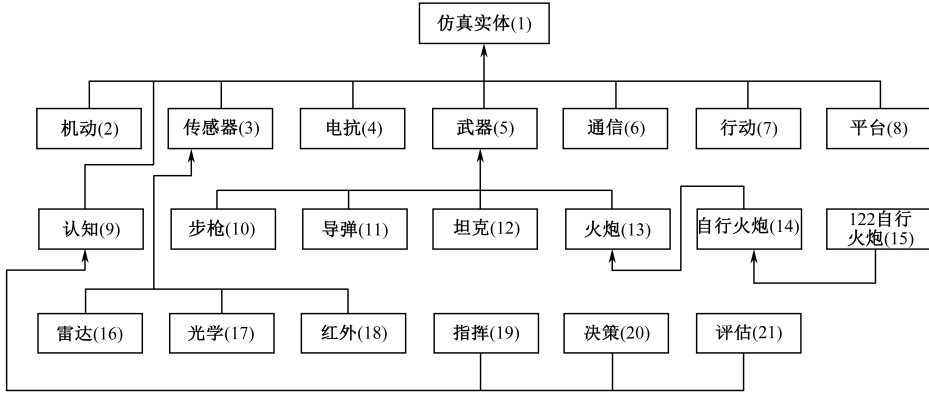


图 3-5 仿真模型体系实例（继承关系）

由图 3-5 可知, 根据模型之间的继承关系构建的模型体系结构为

$$HS = \{M, H\} = A = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

其中, 1 表示两个模型之间有直接继承关系, 0 表示模型之间没有直接继承关系。

根据解析结构模型方法对 HS 进行区域分解，得到可达性集合、先行集合和共同集合，如表 3-1 所示。

表 3-1 模型体系的可达集、先行集和共同集（继承关系）

模型	$L(a_i)$	$D(a_i)$	$L(a_i) \cap D(a_i)$
a_1	$a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13},$ $a_{14}, a_{15}, a_{16}, a_{17}, a_{18}, a_{19}, a_{20}, a_{21}.$	a_1	a_1
a_3	$a_3, a_{16}, a_{17}, a_{18}$	a_1, a_3	a_3
a_5	$a_5, a_{10}, a_{11}, a_{12}, a_{13}$	a_1, a_5	a_5
a_9	$a_9, a_{19}, a_{20}, a_{21}$	a_1, a_9	a_9
a_{13}	a_{13}, a_{14}, a_{15}	a_1, a_5, a_{13}	a_{13}
a_{14}	a_{14}, a_{15}	a_1, a_5, a_{13}, a_{14}	a_{14}

其他模型的 $L(a_i) = \emptyset$ ，因此有共同集 $G = \emptyset$ ，所以，经区域分解 $\pi(HS) = \{p_1\} = M$ ，由此可知，模型体系只能分为一个区，也就是说所有的模型通过继承根类 a_1 构成一个体系。从实际情况看，这符合模型体系的划分规则，基本体现了模型的应用实际。

下面对模型进行区域内级间的分解，除了 $a_1, a_3, a_5, a_9, a_{13}, a_{14}$ 模型之外，其他模型的可达性集 $L(a_i) = \emptyset$ ，所以，对于这些模型一定满足 $L(a_i) \cap D(a_i) = L(a_i)$ ，因此 $L_1 = \{a_i \in p_1 - L_0 \mid L(a_i) \cap D(a_i) = L(a_i)\}$ ，即 $\{a_2, a_4, a_6, a_7, a_8, a_{10}, a_{11}, a_{12}, a_{15}, a_{16}, a_{17}, a_{18}, a_{19}, a_{20}, a_{21}\}$ 。因为 $p_1 - L_0 - L_1 = \{a_1, a_3, a_5, a_9, a_{10}, a_{13}, a_{14}\} \neq \emptyset$ ，所以有

$$L_2 = \{a_i \in p_1 - L_0 - L_1 \mid L(a_i) \cap D(a_i) = L(a_i)\} = \{a_3, a_9, a_{14}\}$$

因为 $p_1 - L_0 - L_1 - L_2 = \{a_1, a_5, a_{13}\} \neq \emptyset$ ，所以还需要继续分解

$$L_3 = \{a_i \in p_1 - L_0 - L_1 - L_2 \mid L(a_i) \cap D(a_i) = L(a_i)\} = \{a_{13}\}$$

$$p_1 - L_0 - L_1 - L_2 - L_3 = \{a_1, a_5\} \neq \emptyset$$

$$L_4 = \{a_i \in p_1 - L_0 - L_1 - L_2 - L_3 \mid L(a_i) \cap D(a_i) = L(a_i)\} = \{a_5\}$$

$$p_1 - L_0 - L_1 - L_2 - L_3 - L_4 = \{a_1\} \neq \emptyset$$

$$L_5 = \{a_i \in p_1 - L_0 - L_1 - L_2 - L_3 - L_4 \mid L(a_i) \cap D(a_i) = L(a_i)\} = \{a_1\}$$

$$p_1 - L_0 - L_1 - L_2 - L_3 - L_4 - L_5 = \emptyset$$

因此，模型可以分为五级 L_1, L_2, L_3, L_4, L_5 ，与图 3-5 中的层次结构是一致的，其中， L_1 中的模型没有子模型， L_2, L_3, L_4 中的模型既有父模型也有子模型， L_5 中的模型没有父模型，只有子模型。采用级间分解的方式，可以了解各模型在体系中的层次及相互间的继承关系。

3.3.4 组合关系建模

以一个简单的仿真实体模型体系为例（见图 3-6），假设该体系中有机动组件、传感器组件、电子对抗组件、平台组件、通信组件、行动组件、认知组件和武器组件等，要形成具有一定仿真能力的模型实体，必然对这些组件进行组合使用才能具备完成作战任务的能力。假设某系统内模型具有组合关系的可达矩阵为



图 3-6 仿真模型体系示例（组合关系）

为了进一步分析模型的组合关系需要对 M 进行区域分解，首先需要找出各要素的可达性集合、先行集合及共同集合，如表 3-2 所示。

表 3-2 模型体系的可达集、先行集和共同集（组合关系）

模型	$L(a_i)$	$F(a_i)$	$L(a_i) \cap F(a_i)$
a_2	a_2	a_2, a_3, a_8	a_2
a_3	a_2, a_3	a_3, a_8	a_3
a_4	a_4, a_5, a_6, a_7	a_4	a_4
a_5	a_5, a_6, a_7	a_4, a_5, a_7	a_5, a_7
a_6	a_6	a_4, a_5, a_6, a_7	a_6
a_7	a_5, a_6, a_7	a_4, a_5, a_7	a_5, a_7
a_8	a_2, a_3, a_8	a_8	a_8

由表 3-2 知 $G = \{a_4, a_8\}$ ，由于 $L(a_4) \cap F(a_8) = \emptyset$ ，所以 a_4, a_8 分别属于两个区域中。另外，由于 a_5, a_6 和 a_7 的可达集合交集非空，所以 a_5, a_6, a_7 和 a_4 在同一区域，故系统可划分为两个区域 $\pi(X) = P_1, P_2$ ，其中 $P_1 = \{a_4, a_5, a_6, a_7\}$ ， $P_2 = \{a_2, a_3, a_8\}$ 。依据区域划分的结构，可将可达矩阵中的要素进行重新排列，得到矩阵 M_H ：

$$M_H = \begin{matrix} & \begin{matrix} 4 & 5 & 6 & 7 & 2 & 3 & 8 \end{matrix} \\ \begin{matrix} 4 \\ 5 \\ 6 \\ 7 \\ 2 \\ 3 \\ 8 \end{matrix} & \begin{bmatrix} 1 & 1 & 1 & 1 & & & \\ 0 & 1 & 1 & 1 & & 0 & \\ 0 & 0 & 1 & 0 & & & \\ 0 & 1 & 1 & 1 & & & \\ & & & & 1 & 0 & 0 \\ & & 0 & & 1 & 1 & 0 \\ & & & & 1 & 1 & 1 \end{bmatrix} \end{matrix}$$

从以上可以看出， M_H 与矩阵 M 不同，在结构上更为清晰。根据表 3-2，按照区域内级间分解的方法对其进行区域间层级分析。

$$L_1 = \{a_i \in P_1 \mid L(a_i) \cap F(a_i) = L(a_i)\} = \{a_i \in a_4, a_5, a_6, a_7 \mid L(a_i) \cap F(a_i) = L(a_i)\} = \{a_6\}$$

即第一级要素为 a_6 ，剩余要素为 $\{a_4, a_5, a_7\}$ ，如表 3-3 所示。

表 3-3 a_4, a_5, a_7 的可达集、先行集和共同集

模型	$L(a_i)$	$F(a_i)$	$L(a_i) \cap F(a_i)$
a_4	a_4, a_5, a_7	a_4	a_4
a_5	a_5, a_7	a_4, a_5, a_7	a_5, a_7
a_7	a_5, a_7	a_4, a_5, a_7	a_5, a_7

$$L_2 = \{a_i \in P_1 - L_1 \mid L(a_i) \cap F(a_i) = L(a_i)\} = \{a_i \in a_4, a_5, a_7 \mid L(a_i) \cap F(a_i) = L(a_i)\} = \{a_4, a_7\}$$

即第二级要素为 a_5 和 a_7 ，剩余要素为 $\{a_4\}$ ，如表 3-4 所示。

表 3-4 a_4 的可达集、先行集和共同集

模型	$L(a_i)$	$F(a_i)$	$L(a_i) \cap F(a_i)$
a_4	a_4	a_4	a_4

$$L_3 = \{a_i \in P_1 - L_1 - L_2 \mid L(a_i) \cap F(a_i) = L(a_i)\} = \{a_i \in a_4 \mid L(a_i) \cap F(a_i) = L(a_i)\} = \{a_4\}$$

即第三级要素为 a_4 ，剩余要素为 \emptyset ，至此，所有要素均被分级，故区域 P_1 共分为三级。

第一级要素为 a_6 ，第二级要素为 a_5 和 a_7 ，第三级要素为 a_4 。采用相同的方法

法可以对区域 P_2 进行分级, 则可得第一级要素为 a_2 , 第二级要素为 a_3 , 第三级要素为 a_8 。根据分解的结果, 将可达性矩阵 M_H 按级变位得 M'_H , 即

$$M'_H = \begin{matrix} & \begin{matrix} 6 & 5 & 7 & 4 & 2 & 3 & 8 \end{matrix} \\ \begin{matrix} 6 \\ 5 \\ 7 \\ 4 \\ 2 \\ 3 \\ 8 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 & & & \\ 1 & 1 & 1 & 0 & & 0 & \\ 1 & 1 & 1 & 0 & & & \\ 1 & 1 & 1 & 1 & & & \\ & & & & 1 & 0 & 0 \\ & 0 & & & 1 & 1 & 0 \\ & & & & 1 & 1 & 1 \end{bmatrix} \end{matrix}$$

从矩阵 M'_H 中知 a_5 和 a_7 相应的行和列的矩阵元素完全一样, 从而可以缩减相应的行和列, 除去 a_7 后得到新的递阶结构分块可达性矩阵 M' :

$$M' = \begin{matrix} & \begin{matrix} 6 & 5 & 4 & 2 & 3 & 8 \end{matrix} \\ \begin{matrix} 6 \\ 5 \\ 4 \\ 2 \\ 3 \\ 8 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & & & \\ 1 & 1 & 0 & & 0 & \\ 1 & 1 & 1 & & & \\ & & & 1 & 0 & 0 \\ & 0 & & 1 & 1 & 0 \\ & & & 1 & 1 & 1 \end{bmatrix} \end{matrix}$$

在区域划分和级间分解的基础上, 分析并建立组合结构模型。求解组合结构模型实际上就是要建立结构矩阵, 这个结构矩阵主要用来反映系统多级递阶结构的问题。下面以一种简化方法分析并找出结构矩阵, 即通过将矩阵 M' 减去单位矩阵 I , 得到新的矩阵 M'' 。这实际上是一个对系统整理而求得的可达性矩阵再还原回去的过程。这种抽象和转换使得对体系的认识更加具体、直观, 并能从结构中直接抽取本质属性和关系。

$$M'' = M' - I = \begin{matrix} & \begin{matrix} 6 & 5 & 4 & 2 & 3 & 8 \end{matrix} \\ \begin{matrix} 6 \\ 5 \\ 4 \\ 2 \\ 3 \\ 8 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & & & \\ 1 & 0 & 0 & & 0 & \\ 1 & 1 & 0 & & & \\ & & & 0 & 0 & 0 \\ & 0 & & 1 & 0 & 0 \\ & & & 1 & 1 & 0 \end{bmatrix} \end{matrix}$$

在矩阵 M'' 中, 先找一级与二级之间的关系, 再找二级与三级之间的关系,

直到将每一个分区的各级找完为止,则可得到结构矩阵 A' 。从 M'' 中知 $m''_{56}=1$, 说明节点 a_5 与处于第一级的节点 a_6 有组合关系,即 $a_5 \rightarrow a_6$, 然后抽去 a_6 的行和列再找第二级与第三级之间的关系,又知 $m''_{45}=1$, 说明节点间 a_4 与 a_5 有组合关系。以此类推,可以将 P_2 区域中的节点间关系也找出来, $m''_{32}=1$, $m''_{83}=1$, 最后将 $m''_{56}=1$, $m''_{45}=1$, $m''_{32}=1$, $m''_{83}=1$ 作为结构矩阵的元素,画出结构矩阵 A' 如下:

$$A' = \begin{matrix} & \begin{matrix} 6 & 5 & 4 & 2 & 3 & 8 \end{matrix} \\ \begin{matrix} 6 \\ 5 \\ 4 \\ 2 \\ 3 \\ 8 \end{matrix} & \begin{bmatrix} 0 & 0 & 0 & & & \\ 1 & 0 & 0 & & 0 & \\ 0 & 1 & 0 & & & \\ & & & 0 & 0 & 0 \\ & 0 & 1 & 0 & 0 & \\ & & & 0 & 1 & 0 \end{bmatrix} \end{matrix}$$

有了结构矩阵 A' , 就可以绘制出体系的多级递阶有向结构图如图 3-7 所示。

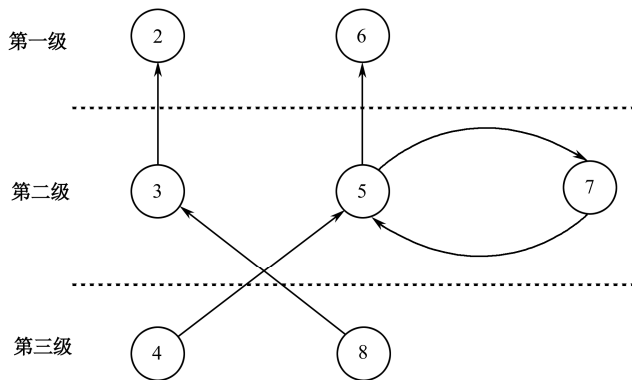


图 3-7 多级递阶有向结构图

从图 3-7 可以看出: 机动模型、传感器模型和认知模型可以按照一定方式建立组合关系, 并形成更加复杂的组合模型。以我方某型飞机作战过程仿真为例, 假设我方某型飞机在飞行(机动模型)过程中, 通过机载雷达(传感器模型)发现了一个可疑飞行目标, 模型系统中的自主决策模型(认知模型)根据雷达侦察到的各种参数, 并与敌方作战飞机性能数据库进行比对, 判定各种飞行参数符合敌方某型飞机的特征。一旦敌方飞机进入到我方飞机的攻击范围, 则进行

自动锁定和攻击。因此,在仿真过程中如此复杂的过程可以通过这样一个组合的复杂模型直接完成。另外,采用同样的方法可以对聚合关系、指控关系和交互关系等进行形式化建模,这里不再一一赘述。

3.4 可组合的模型体系层次结构分析

模型体系结构是创建模型部件一致性的视图集合,通过一致性的描述形式来满足模型开发人员、模型使用人员和模型管理人员的需要。构建可组合的模型体系是为了使模型能在更大范围内重用和组合,因此,模型体系结构的作用体现在以下几个方面:

(1) 模型体系是模型系统建设中的关键环节,对模型的设计、开发、集成、组合和重用等起重要的指导作用。

(2) 模型体系以一组相互作用的功能部件来描述模型系统的构成和关系,围绕战场中的作战编组、武器装备、作战行动和战场环境等进行建模,能够对战场空间各类对象、事件、作用过程和交战关系等进行抽象和描述,便于军事人员加深对作战规律的认识和理解。

(3) 通过采用层次化、模块化等方法构建模型体系,能够使模型体系具有良好的可扩展性、可维护性和可组合性,便于模型开发和应用人员能够直观把握模型之间的组合关系、继承关系和指挥控制关系等^[32]。

(4) 以体系的方式将模型进行分类管理,使模型成为在特定仿真环境下相互作用的由若干实体组成的整体,整体可以呈现出新的功能,这种整体功能是模型在独立状态时所不具备的。

(5) 以模型体系的形式组织和管理模型,对模型本身提出了新的要求,需要对模型进一步抽象,特别是对于具有相同或相似功能的模型部件而言,需要采取抽象的方法对共同的要素和关系进行描述。一方面可以减少模型间的重复性,另一方面可以保证模型描述的一致性,并通过组件组合、代码组合和参数设置等手段在多个层次上以组合的方式开发更加复杂的模型。

综上所述,模型体系一方面要能客观描述和抽象战场空间实体,另一方面要能充分体现模型在功能和结构方面的可组合性,以便实现模型的重用和组合。因此,模型体系需要遵循以下几个原则。

(1) 具备模型描述的能力。即能够完成对战场空间中认知模型、环境模型、平台模型、机动模型、武器模型、传感器模型、行动模型、通信模型、电子对抗模型及其他的一些辅助模型的描述。

(2) 提供模型重构和组合的技术框架。模型重构的基础是具有独立功能的模型组件，这些组件之间不存在重复部件，并且拥有可以相互组合的标准接口。

(3) 提供合理的模型分类和管理功能。不管是按照兵种专业还是功能类别对模型分类，最基本的要求就是不同的模型之间不能存在重复或包含关系的部件。以预警机模型为例，预警机是一种整体模型，其中可能由机动模型、雷达模型和通信模型组成，在对预警机模型建模时不需要具体描述雷达模型的功能，而是预留雷达模型的组件式接口。为避免重复，实际上预警机模型是通过多个具有独立功能的模型组件组合而成的，这样在方便模型管理的同时提高了模型的重用性。

采用自底向上建模方法，首先分析作战仿真应用需求，通过对作战过程的抽象，形成一系列基本组件，这些基本组件可以看作是原子组件。将这些原子组件通过标准的接口相互组合后，形成具有一定作战仿真能力的功能组件模型，然后通过这些功能组件的组合形成更加复杂的实体模型。

根据上述建立模型体系的原则，提出面向服务的可组合模型体系的结构模型。为了提高模型的重用性，从模型组合的角度出发，面向服务的可组合模型体系由三个模型层构成，分别是基本组件层、功能组件层和仿真模型层，如图 3-8 所示。

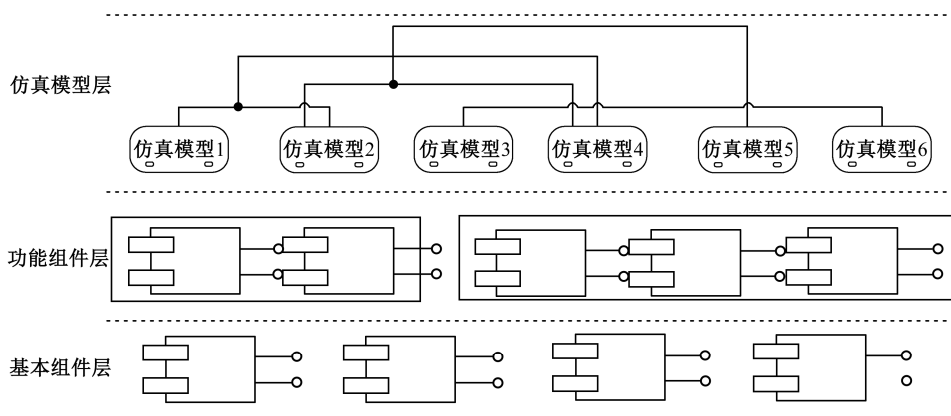


图 3-8 可组合的模型体系层次结构

第一层是基本组件层。基本组件层由战场空间实体模型和软件组件构成,属于原子组件,不具备独立仿真能力。基本组件是功能组件和模型服务的根类,它是对战场空间实体的公共功能和属性的描述,通过标准的接口,提供组合的能力。功能组件和模型服务都需要通过基本组件实现对模型的重构和组合。对于模型服务而言,可以通过组合功能组件实现仿真功能,还可以组合不同的模型服务实现更为复杂的仿真功能,因此模型服务可以提供更加灵活的组合建模能力。

第二层是功能组件层。功能组件层由作战仿真的基本功能组件组合构成,是通过基本组件组装后得到的具有一定模型能力的组件,是对作战过程中涉及的各种对象、行为、事件和活动的描述。功能组件在实现这些基本功能时,具备相对稳定的能力,通常不随仿真系统的变化而变化;而模型服务层会因仿真系统的需求而变化,这种需求的变化通过服务的组合实现。由于功能组件层的存在,服务模型可以通过功能组件组合而成,从而满足不同仿真应用的需求。

第三层是仿真模型层。它描述的是仿真应用模型,也就是可以供仿真系统直接访问和调用的仿真模型,以服务的形式对功能组件或组合后的模型进行封装,利用面向服务组合的方法,对外提供模型服务,使组件模型按照不同粒度的组合重用。根据实际仿真系统的需求、组合形成不同粒度、不同层次的模型服务,满足不同仿真系统的要求,在更大范围内、更高层次上实现有效的重用。

在这个三层结构模型中,仿真模型层对应于作战仿真实体,功能组件对应于仿真实体的子功能,基本组件层为模型的组件化模块。这三层相互联系与制约,仿真模型层是对基本组件层和功能组件层进行聚合的结果表现,即在应用过程中是按照自底向上的方式聚合形成不同层次、不同分辨率的仿真模型;基本组件层和功能组件层是对仿真模型层按照不同的粒度进行解聚后的结果表现,即在模型分析与设计过程中,以自顶向下的方式解聚后,按照基本组件和功能组件的形式对模型进行建模的。因此,这种模型体系能够分别解决模型开发和应用的问题,使模型开发人员和应用人员能够直观地理解模型及模型之间的关系。

模型体系除了包含模型集合外,还需要描述模型之间的关系。这种模型体系结构除了能够对模型进行很好的分类和管理外,对于模型之间的关系,如组合关系、继承关系、隶属关系和交互关系等也能有很好的描述。

另外,作战仿真模型体系不应该是固定不变的,应具有一定的扩展性和继承性。组成体系的基础是组件,通过对组件的完善,可以不断丰富模型体系,以适应不断发展过程中的作战仿真需要。

3.5 本章小结

本章首先介绍模型及模型体系的基本概念,阐述了模型体系构建过程中面临的几个重要问题。以 JWARS 和 FLAMES 仿真系统的模型体系为例,分别从战役级与任务级两个层次,分析模型体系的模型组合功能与设计方法,并建立了符合作战实际的模型组合的层次结构。采用解释结构模型对模型体系中的继承关系、组合关系进行了形式化建模,并构建了可组合的模型体系框架。

第 4 章

基于组件的作战仿真模型及 体系设计



基于组件的建模技术不但可以降低模型间的耦合度，还能提高模型的重用性，使模型开发与应用相互独立，实现接口与逻辑分离^[33]。组件化设计是模型重用的重要方法，组件化方法用标准的接口规范描述模型所能提供的功能，将业务功能与平台分离，消除平台对模型功能的影响，实现模型的平台无关性。

4.1 军事概念模型设计

随着建模仿真技术的发展，军事建模与仿真越来越受到各级领导的重视，军事概念模型的提出从一定程度上消除了问题域与求解域之间的差异，为军事人员和技术人员对作战问题的一致性理解提供了很大的帮助。但由于技术机制和管理体制等原因，各科研单位对概念模型的重视程度不一，还存在诸多问题：①概念模型描述缺乏统一的标准。特别是对于大型复杂系统而言，不同的单位可能承担不同的模型开发任务，在模型对接时，往往会存在很多描述上的偏差。②概念模型设计方法不一致。由于存在多种概念建模工具，每一种工具对同一问题的描述的表现形式不同，使得概念模型在语法和语义上缺少一致性，概念描述存在二义性。③概念模型内容不全面。对于同一类型模型而言，不同的兵种单位描述角度、

描述属性和建模元素不同,使得模型在组合过程中出现逻辑关系不清晰、数据流不明确等问题^[34]。因此,有必要规范军事概念建模过程,明确建模内容和固化概念模型描述模板。

4.1.1 军事概念模型建模思路

通过作战仿真系统实现对战场环境、战斗进程及作战方式的模拟,这是一种较为科学的作战研究手段,通过定量与定性相结合的方式深化对作战规律的研究。而实现上述目标的基础条件是,首先构建一个层次合理、内容全面、易于调用和组合的模型体系,这个体系应包括满足不同作战行动任务的仿真。在这个模型体系中,每一个模型都由军事概念、数学逻辑和程序模型三个部分组成,这也是仿真建模的三个重要阶段^[35]。

对作战过程的仿真实际上是对作战任务空间的具体抽象,可将作战的实际过程、人的指挥行为、武器装备的功能及作战方法的运用等以计算机模型的形式呈现出来。军事概念模型是对作战问题从现实世界到仿真世界的具体体现,因此军事概念模型起着纽带作用,对于军事指挥人员和仿真技术人员的沟通与协作十分必要,能够提高仿真的正确性、重用性和互操作性。

军事概念模型体系是从作战任务空间的总体要求出发,对参战力量、主要行动和作战空间做出的整体规划。军事概念模型是军事人员依据条令、条例和军事经验,面向仿真建模领域,用军事人员习惯的军事语言,对各类基本作战实体、作战行动、作战环境、影响因素,以及各类实体与实体之间、行动与行动之间、条件与行动之间相互作用的规则化军事描述,是一种结构化或半结构化的抽象描述。其目的是为建立逻辑数学模型和程序仿真模型提供军事依据,确保军事与技术人员对同一军事问题理解的一致性。

军事概念模型建模分为面向领域和面向设计的概念建模,只有将领域问题弄清楚后才能完成模型设计。其基本步骤分为军事模型需求分析、军事概念模型描述、军事概念模型抽象和军事概念模型表示,如图 4-1 所示。

军事模型需求分析包括军事问题分析、建模任务分析和模型功能需求分析等主要内容。军事问题分析是要将仿真的军事问题进行明确的抽象,并对实际的军事问题进行具体化和模型化描述。建模任务分析主要是根据所要描述的军事问题构建模型任务清单,明确建模的目的、要素和内容。模型功能需求分析主要是依

据建模任务对模型所要实现的功能进行具体描述,包括定性描述与定量描述。定性是从总体上对模型所要实现的目标进行概括性描述,定量则是从子功能的数量、内容及其与其他模型之间的关系进行细化描述。

军事概念模型描述主要分为模型实体描述、模型关系描述和模型行为描述。模型实体描述是对不同层次不同类型实体的区分,将需求描述中的实体类型等相关信息进行梳理。模型关系描述是在对行动实体在执行特定战斗动作过程中,与其他实体之间和实体动作之间关系的总体描述,目的是对行动实体在不同层次使命空间中所涉及的关系类型进行梳理。模型行为描述是对特定行动实体完成指定作战行动的抽象描述。

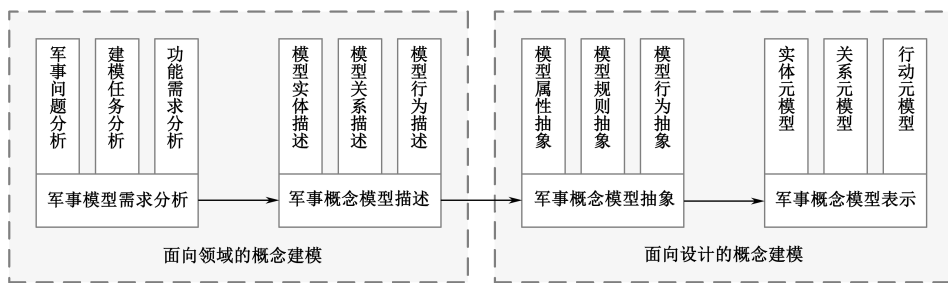


图 4-1 军事概念建模基本步骤

军事概念模型抽象主要分为模型属性抽象、模型规则抽象和模型行为抽象等。军事概念抽象主要是通过对作战编组、作战阶段、主要作战任务行动及作战环境等分析,对模型属性、规则和行为等进行描述。可用类图描述作战过程中的实体及其之间的静态关系,用活动图描述实体、动作及任务之间的动态关系,用交互图描述出各实体间的信息交互流程^[36]。

对于军事概念模型表示,本书主要采用 EATI 方法对作战过程进行抽象描述,形成实体元模型、关系元模型、行动元模型和交互元模型等,并通过填写概念模型模板对描述对象及其动作过程和军事规则给出一个集中和规范化的描述。

4.1.2 基于模板的概念模型描述与设计

军事仿真概念模型是一种独立于具体仿真实现的表示,是对作战过程中的实体、动作、关系和属性等要素的抽象描述。本节提出一种基于模板的军事概念模

型建模方法（Military Concept Model based on Template），主要从原子角度对概念模型进行划分，通过实体元模型、动作元模型、关系元模型、属性元模型和数据元模型等对军事概念模型进行描述。

1. 实体元模型描述

实体元模型（ENM）描述主要是对模型的名称、类别、编号、功能、所属兵种、应用层次等信息进行描述，用六元组表示：

ENM:: = <ENS, ETS, ECS, EFS, EAS, ELS>

其中，ENS——名称段，表示模型的名称，与模型 ID 绑定；

ETS——类别段，表示模型所属的专业类别；

EFS——功能段，用来说明模型的主要用途；

ECS——编号段，是模型的 ID，通过编码对模型进行唯一性标识；

EAS——军兵种段，模型所属的军种、兵种；

ELS——应用层级段，说明模型的粒度。

以火箭炮模型为例，其模板描述如表 4-1 所列。

表 4-1 实体元模型描述示例

元模型名称	火箭炮
类别	射击类
功能	火力覆盖
编号	按规则命名
军兵种	陆军、炮兵
层级	单武器

2. 动作元模型描述

动作是由实体模型执行的，主要对模型主体所涉及的所有动作行为及其功能进行描述。动作元模型可以表示为如下的八元组：

ACM:: = <ACES, ACSS, ACNS, ACCS, ACFS, ACSS, ACOS, ACCS>

其中，ACES——执行实体段，描述执行动作的主体名称和编号（与实体模型中的 ENS 关联）；

ACSS——动作集合段，描述所能完成动作的类型列表，如进攻、射击和机动等；

ACNS——名称段，赋予每一个动作的名称；

ACCS——编号段，与名称相对应的动作编号，即与名称绑定的动作 ID；
ACFS——功能段，描述每一个动作执行所能完成的具体功能；
ACIS——输入段，描述动作的触发条件，即执行动作的输入信息；
ACOS——输出段，描述动作执行后的输出信息；
ACCS——控制段，根据作战条令，对所有作战动作进行规划；
以坦克模型为例，其动作元模型描述如表 4-2 所示。

表 4-2 动作元模型描述示例

执行实体 名称	坦克排				
动作集合	名称	编号	功能	输入	输出
	机动	T001	位置移动	时间、目标位置	路径信息
	冲击	T002	快速冲向攻击目标	冲击时间、路线、 目标区域	攻击单位的位置及当前状态
	射击	T003	集中火力摧毁目标	射击时间、方向、 目标位置	坦克排位置、状态及敌方毁 伤信息
控制	机动 规则	遇障绕行规则、遇敌袭击规则、机动速度调整规则等			
	冲击 规则	遇敌打击时的处理规则、交火规则、冲击停止规则等			
	射击 规则	战斗力指数计算规则、目标取舍规则、有效射击规则等			
	时序	执行时序按照具体的行动任务			

3. 关系元模型描述

关系元模型描述以行动为中心的交互关系，交互关系区分为实体间的关系、实体与动作间的关系、动作与任务间的关系及任务间的关系四大类，如通信关系、指挥关系、配属关系、控制关系、供给关系、对抗关系等。上述关系经过统一抽象后变成两种关系，即前驱交互关系和后继交互关系。关系元模型可用五元组表示为

$$REN M::= \langle RNS, RTS, RAS, RIS, RDS \rangle$$

其中，RNS——名称段，包括前驱、后继交互关系的名称和编号；

RTS——类型段，说明前驱、后继交互关系的类型，如指挥关系、配属关系等；

RAS——动作段，包括前驱、后继交互关系涉及的动作名称和类型；

RIS——接口段，描述与其他模型交互时的输入、输出接口的名称与编号；

RLS——逻辑执行段，主要描述关系执行的时间顺序、逻辑顺序等。

以行动为中心的关系元模型描述如表 4-3 所示。

表 4-3 关系元模型描述示例

元模型名称	前驱关系名称: ** 编号: ** 后继关系名称: ** 编号: **
类别	前驱关系类型: ** 后继关系类型: **
动作	前驱动作名称: ** 类型: ** 后继动作名称: ** 类型: **
接口	输入接口名称: ** 编号: ** 输出接口名称: ** 编号: **
逻辑	关系执行的时间顺序、逻辑顺序（并发顺序）

4. 属性元模型描述

属性元模型（ATTM）描述主要是对模型各种属性进行具体描述，主要分为标识属性、实力属性、动态属性、任务属性、效能属性等。采用五元组表示：

ATTM::=<ATIS, ATOS, ATDS, ATTS, ATES>

其中，ATIS——标识属性段，主要描述模型所属方、级别、对应单位编制名称和代字；

ATOS——实力属性段，模型配备人数、装备数、上级单位和作战编成名称等；

ATDS——动态属性段，描述模型的行动方式、空间位置、生存状态和行动方向；

ATTS——任务属性段，描述具体执行的任务内容、任务等级和任务执行情况的状态；

ATES——效能属性段，描述的是模型的火力指数、机动能力指数、防护能力指数、战斗力指数和信息共享能力指数等。

以机械化步兵连模型为例，其模板描述如表 4-4 所示。

表 4-4 属性元模型描述示例

实体名称	属性类型	属性名称	属性取值
机械化步兵连	标识属性	所属方	1.红方；2.蓝方；3.不明方
		级别	连
		单位编制名称	××团×营×连
		单位代字	固有编号

续表

实体名称	属性类型	属性名称	属性取值
机械化步兵连	实力属性	人员编制数	110 人
		武器装备名称	步兵战车
			班用轻机枪
			120 火箭筒
		上级单位名称	陆军×机步团×营
		作战编成名称	左翼突击群
	动态属性	行动方式	摩托化机动
		空间坐标	军事坐标值
		生存状态	完好
		配置或行动方向	××方向集结地域
	任务属性	任务内容	战术机动、战术包围、抗击反冲击
		任务等级	任务需求等级
		状态标志	表明任务执行的状况
	效能属性	火力指数	攻击能力
		机动能力指数	机动能力
		防护能力指数	综合防护能力
		战斗力指数	由综合作战能力确定
		信息共享能力指数	高、中、低

5. 数据元模型描述

数据元模型（DAM）是对模型属性定量数据的描述，主要分为位置数据、战斗力指数数据、速度数据、时间数据、装备数据、人员数据、通信距离数据、战场防护数据、弹药量数据和战损数据等。采用以下十元组表示：

DAM::=<DLS, DIS, DVS, DTS, DES, DPS, DCS, DFS, DAS, DDS>
其中，DLS——位置段，主要描述模型的坐标和高程信息；

DIS ——战斗力指数段，主要描述模型针对不同目标的火器指数；

DVS——速度段，描述模型的运动速度；

DTS——时间段，描述模型执行任务、完成动作的各种时间；

DES——装备数量段，描述模型配备装备的数量；

DPS——人员数量段，主要描述模型配置的人员数量；

DCS——通信距离段，模型与其他实体通信的有效距离等；
DFS——战场防护段，描述模型的战场防护能力；
DAS——弹药量段，描述模型携带的弹药数量；
DDS——战损段，描述模型的战损情况。
以坦克连模型为例，其模板描述如表 4-5 所示。

表 4-5 数据元模型描述（坦克连模型）

模型名称	坦克连
位置数据	坐标（35000，23456），高程 10m
战斗力指数数据	打步兵火器指数 1600 打装甲目标火器指数 3200 打空中目标火器指数 800
速度数据	30~60km/h
有效射程	30km
时间数据	机动时间、射击时间、攻击时间等
装备数据	10 辆主战坦克 3 辆特种坦克
人员数据	45 人
战场防护数据	85
弹药量数据	120 发
无线通信距离	25km
战损数据	5 人
展开幅员	0.2km ²

元数据属性注册代码如下：

```
BEGIN_METADATA( DATAMETAMODEL)
  REG_BASE(DAM)
  REG_PROP(LocationData, “位置数据” )
  REG_PROP(StrengthIndexData, “战斗力指数数据” )
  REG_PROP(Velocity Data, “速度数据” )
  REG_PROP(TimeData, “时间数据” )
  REG_PROP(Equipment Data, “装备数据” )
  REG_PROP(PersonData, “人员数据” )
```

```
REG_PROP(CommunicationDistanceData, “通信距离数据” )
REG_PROP(FenseData, “战场防护数据” )
REG_PROP(Ammunition Data, “弹药量数据” )
REG_PROP(DamageData, “战损数据” )
END_METADATA()
```

6. 作战实体、行动、元动作模型的关系模型

陆军作战模型中作战实体、行动与元动作模型的相互关系如图 4-2 所示，通过该图能够简要说明实体是如何按照行动计划完成相关动作的。

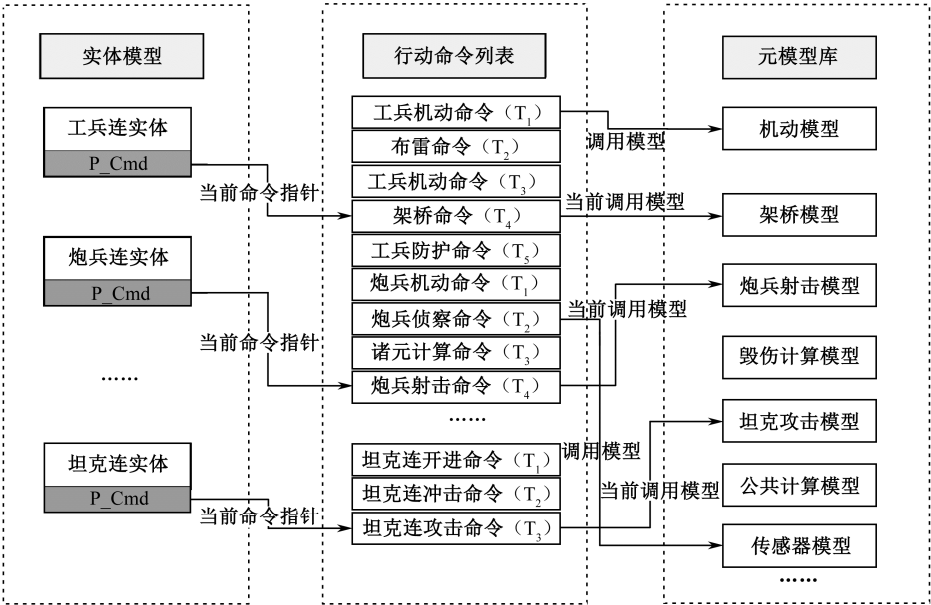


图 4-2 实体、行动与元动作模型关系

此处主要采用作战实体与行动关联的方式实现对模型的驱动，即作战实体按照上级赋予的作战任务形成若干个作战行动，以行动计划列表的形式发布行动命令，而后根据作战任务对命令按照专业、行动类别进行区分，对作战行动计划进行分解后，以元动作的方式分别调用射击、进攻、公共计算和环境等组件模型完成相应作战任务。

1) 元模型库

元模型库是作战仿真模型系统的核心,主要是陆军部队各兵种主要作战实体的基本行动的集合。这些基本行动模型作为元模型向各作战实体提供各种可能的组合模型。因此,元模型库是由各种元动作模型构成的,是对一些模型的共同部分进行抽象而形成的,包括机动、进攻、射击、布雷、开进、侦察和防护等基本模型,还可以结合公共计算模型,如通视计算、地形计算、距离计算等模型。对这些模型的调用是按照一定的时间序列进行的,从而使模型按照格式化命令和时间戳运行^[37]。

2) 行动命令列表

行动命令列表的主要功能是按照计划执行的时间顺序,集中保存指挥员所制定的作战行动计划数据。这些数据是一些格式化命令,系统能够自动解析。同时,这些命令有严格的时间戳定义,这种时间戳使得模型能够按照下达作战命令的时间推进整个作战进程。在一次作战行动计划中可能会包含多条行动命令,如机动、侦察、防护攻击任务等,将其分解后统一存放在行动计划列表中统一管理。行动命令列表是一个动态管理的数组,以图 4-2 为例,存放了用户针对工兵连、炮兵连、坦克连的三条计划任务。这三条计划任务分别被拆分为 5、4、3 条元动作,通过实体模型中的当前命令指针的关联,达到实体模型按照时间顺序执行作战计划的命令。

3) 行动实体

行动实体通过“P_Cmd”指针激活格式化命令,指针指向某一条格式化命令,说明这条行动命令即将被执行。根据不同格式化命令的要求,输入相应的参数完成命令下达任务,形成实体—格式化命令—调用模型—执行模型—推进战斗进程的完整环节,实现对模型的统一管理。

4.2 作战仿真模型组件设计

组件伴随着面向对象方法的发展而产生,组件化方法是实现重用的基础。通过标准、规范的接口与外界产生交互,外部对组件的调用通过组件接口完成,无须关心组件内部的具体实现方式,只需关心访问接口的相关参数。模型组件的继

承性使其能从抽象的模型中抽取复用部分,通过组件的相互组合形成具有更复杂功能的模型组件。

4.2.1 组件设计原则与要求

针对陆军部队作战能力的特点与发展的目标和要求,结合组件建模的原则,作战仿真模型组件设计须遵循以下要求:

(1) 针对性原则。体现陆军部队作战能力的特点,符合陆军部队发展的目标和要求。陆军部队的武器装备系统和体制编制都不同于常规作战部队,有其自身的特点,其作战能力涵盖的范围很广,模型组件几乎涉及作战过程的各个环节,作战仿真模型组件应能够体现陆军部队的编成、系统、指控关系和装备特点等。

(2) 全面性和科学性原则。全面性即所构建的模型组件应能涵盖陆军部队作战的各个因素,并且在信息化条件下,尽量简化模型组件的同时确保组件的科学、合理性,即模型体系的设计力求科学、准确地反映陆军部队作战行动特征。

(3) 可行性和可操作性原则。陆军部队作战仿真模型组件应反映部队发展现状和趋势,设计的模型组件应具有可描述性和可量化的特点。只有准确且直观模型组件才便于程序设计人员的使用,但同时要确保模型的军事合理性。

(4) 定性分析与定量分析相结合的原则。模型组件的最终结果是提供给程序人员进行仿真实现,因而,必须将反映陆军部队作战行为的定性指标进行定量化、规范化,为最终的程序实现奠定基础。

4.2.2 组件描述

在系统仿真领域,随着仿真系统的日益复杂,仿真应用开发的工作量也成倍增加,完全从头开发仿真系统降低了工作效率,模型与仿真结果的可靠性与准确性难以得到保证。将分层次的模型以组件的形式进行封装,使得模型库在组织形式上表现为可重用的模型组件库^[38]。可重用的模型组件库不同于以前的模型对象库,模型组件库保存的是一些经过测试的模型组件。这些模型组件遵从标准或规范,并且,模型组件库中还应包括这些模型组件的细节功能描述文档,以实现共享模型服务。

通过对 FLAMES、JWARS 和 SPEEDES 等美军较为典型的仿真系统的模型

研究,可以进一步了解其模型的实现机理、组合运行方式、模型交互方法及模型间进行数据传递的过程等,为模型重用和模型组合设计提供基础^[39]。

FLAMES 仿真模型:

```
FLAMES_SCM = < EquipmentModel, RecognitionModel, Message >
EquipmentModel =< Attri, Method, RefEntity, Message >
RecognitionModel =< ProcessMethod, MessageProcessMethod, Message >
```

FLAMES 仿真模型体系主要由装备模型和认知模型组成,在一个仿真组件中可能包含多个认知模型和装备模型。认知模型具有对装备模型进行管理的功能,能够对装备模型的运行时间、方式及其过程进行控制,采用过程方法实现对指挥和决策活动的仿真,采用消息的方式触发装备模型启动相应的功能执行具体的行动,并与其他与之相关的组件提供模型数据。在这一过程中,装备模型也可能需要引用其他组件或实体,主要是通过 FLAMES 内核获取相关引用句柄,同样采用消息机制实现与其他组件或实体的交互。从本质上看,FLAMES 的组件单元本身就采用了组合思想,在组件层采用组合的方式完成基本功能,因此,整个 FLAMES 模型体系在模型组合方面具有很好的灵活性。

JWARS 仿真模型: *JWARS_SCM = < Attri, Method, SubMethod, RefEntity, Event > SubModel = < PlatformModel, WeaponModel, CommunicationModel, RecognitionModel, ... >* JWARS 的仿真模型主要由模型属性、调用方法、各种子模型、实体引用和事件驱动等组成,即通过对模型状态的描述,基于事件驱动的方式,通过对其他实体的引用,根据仿真任务需要,将平台模型、武器模型、通信模型、感知模型和行动模型等战场空间实体按照一定的规则和方法进行组合运用,以实现不同粒度模型的功能。仿真引擎通过实体引用的方式为战场空间实体提供仿真模型数据,模型之间的交互通过事件驱动的方式完成。

SPEEDES 仿真模型: *SPEEDES_SCM = < Attri, Method, SubEntity, Object Proxy, Event >*, SPEEDES 的仿真模型与 JWARS 仿真模型比较相似,由模型属性、方法、子实体、对象代理和事件构成。模型属性实际上是对其状态的描述,仿真模型通过对子实体组合的方法实现复杂模型的仿真应用。在这个过程中,通过采用对象代理等方法获取其他模型实体的属性,并能及时更新和反馈仿真实体的属性值变化。模型之间的交互通过发送事件完成,每个事件有唯一的句柄标识,通过获取句柄就能获取处理事件的方法,基于事件系统完成对仿真引擎的驱动和管理。

通过对以上三种类型的仿真模型的分析,得出以下几个结论:①描述模型的

状态通过其属性进行表征,基于面向对象的建模和设计方法,通过方法的调用实现模型的仿真运行功能;②模型具有组合、重用功能,在组合方式上,均是基于原子模型或原子实体进行模型组合,采用消息驱动或事件驱动的方式完成模型的组合运行;③模型通过对象代理和实体引用获取其他相关模型的属性值,并以消息驱动或事件驱动的方式完成对其他仿真实体属性值的更新;④模型之间能够按照相应的机制、协议进行数据交互和功能交互;⑤模型本身并不具备智能、自主驱动的功能,模型调用是被动的,必须有人为指令或内部计算机指令通过仿真引擎完成模型事件或任务规划,并按照固有指令驱动模型执行。书中采用基于组件的仿真模型设计方法。

1. 组件模型

```
Sim_CM=<Attributes, Operations, Input, Output, State, Event,  $\Pi$ >
Attributes=<Name, ID, Function, Type, Date, Level, Version>
```

模型采用组件的方式进行组装,因此组件是模型的最小单元,仿真模型组件采用七元组的方式进行描述,主要由属性(Attributes)、操作方法(Operations)、输入(Input)、输出(Output)、状态(State)、事件(Event)和约束条件(Π)构成。其中,Attributes是关于组件模型的所有属性的描述,包括动态属性和静态属性两个方面,如名称、标识、功能和分类类型、建模时间、模型层级和版本等;Operations是对组件所具有的操作方法的描述,是对组件所具备的能力描述和抽象,对组件操作的方法可能有多种,组件的行动能力或完成任务的能力由操作方法具体实现;Input是对输入端口的描述,主要用于接收和处理其他组件或实体发送的消息,进行接口的交互和识别;Output是对输出端口的描述,主要用于向其他组件或实体发送消息,组件与其他实体或组件之间的通信需要按照输出接口的相关标准进行,完成数据的交换;State反映了组件当前的状态,是否可能,是否为智能组件等;Event是驱动组件运行的触发手段,通过事件驱动模型组件的运行。 Π 是一些布尔表达式的集合,它是确保组件在生存周期内需要满足的条件,采用规则的方式进行描述并进行条件判断。组件模型的XML描述如下:

```
<xs:Sim_CM name="组件模型">
  <xs:attributes>
    <xs:element name="名称" type="xs:string"/>
    <xs:element name="标识" type="xs:string"/>
    <xs:element name="功能描述" type="xs:string"/>
```

```

        <xs:element name="类型" type="xs:string"/>
        <xs:element name="建模时间" type="xs:datetime"/>
        <xs:element name="分辨率" type="xs:string"/>
        <xs:element name="模型版本" type="xs:string"/>
    </xs: attributes >

    <xs:attributeGroup ref=" Sim_CM:方法"/>
    <xs:attributeGroup ref=" Sim_CM:输入"/>
    <xs:attributeGroup ref=" Sim_CM:输出"/>
    <xs:attributeGroup ref=" Sim_CM:状态"/>
    <xs:attributeGroup ref=" Sim_CM:事件"/>
    <xs:attributeGroup ref=" Sim_CM:条件"/>

</xs: Sim_CM >

```

2. 组件模型事件

Event=<Time,ID,Port,Value,State,EventType,Sender,Receiver,MessageQueue,Mission,IT>

事件主要有发生的时间、事件标识、执行事件的端口、返回值、当前状态、事件驱动方、事件被驱动方、事件消息队列、任务和触发事件的基本条件。

Mission=<Time,ID,MissionType,Entity,Action,Condition>

其中，任务由时间、任务标识、任务类型、参与实体、具体行动和执行任务的条件等组成。

3. 组件模型端口

Port=<Name,ID,Type,Successor,PortQueue,Precursor,Owner,Condition>

Port 采用七元组的方式进行描述，Name 是对接口名称的描述，接口名称直接关系到数据的传递，在进行数据传输过程中，需要将所传数据类型的名称与接口名称一致表达，实现数据接收。ID 表示接口的标识符，数据传输过程中，除了要与接口名称匹配外，还能判断接口的 ID，只有对这两个属性进行正确判断后才能完成数据传递。Type 表示接口的类型，接口的类型主要用于对接口的分类，通过构建分类树，使得在接口名称查询时能快速定位。Successor 表示接口的后继。以组件 A 为例，A 的后继接口实际上是一个接口集，此集合内包含与组件 A 中 Successor 输出数据的其他所有接口。由于组件 A 可能要与多个接口建立交互关系，因此 A 中 Successor 与其他接口的连接有一个排列关系及交互的先

后关系 PortQueue。Precursor 是对前导接口的描述,是其他组件与组件 A 接口的所有其他接口的集合,也通过排列关系及交互的先后关系 PortQueue 进行描述。Owner 表示接口的拥有者,即仿真组件。仿真组件会为它公布或订购的每个对象类和交互类创建一个同名的输出或输入端口。Condition 是一些布尔表达式的集合,体现接口在生存周期内需要满足的条件,即接口能够进行相互访问的触发条件。

4.2.3 组件模板与视图设计

按照仿真模型建模阶段和系统开发使用流程,战场空间中作战实体的建模过程可划分为三个阶段:组件模板开发阶段、组件模型配置阶段和实体模型组装阶段。每阶段建模工作输出的产品分别为^[40]组件模板、组件模型和实体模型,主要过程如图 4-3 所示。

组件模型是构成模型体系的基础,下面重点对组件模板和组件模型进行介绍。

1. 组件模板

组件模板对武器装备/作战系统中最小可分辨单位进行建模。按照建模对象的功能类别,组件模板分为平台组件模板、通信组件模板、机动组件模板、武器组件模板、认知组件模板、行动组件模板、传感器组件模板和辅助组件模板等几大类,如图 4-4 所示。每类组件模板又可以细分,用户可根据需要对其进行扩充。每个组件模板都有相应的最基础的继承基类。

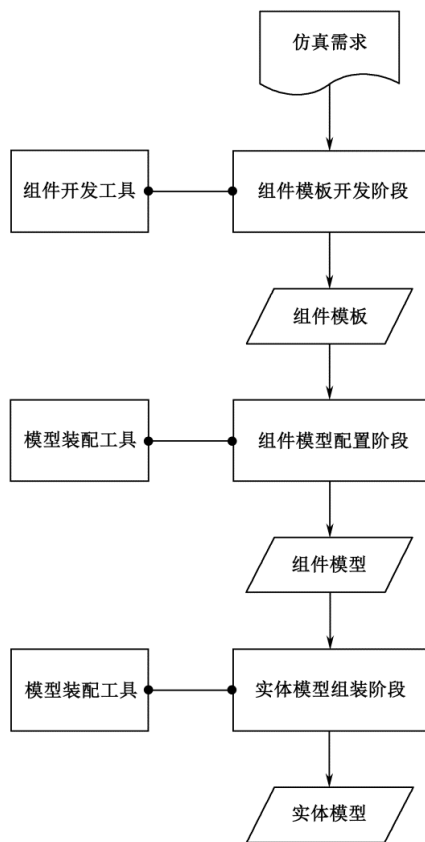


图 4-3 组件模板、组件模型与实体模型的开发过程

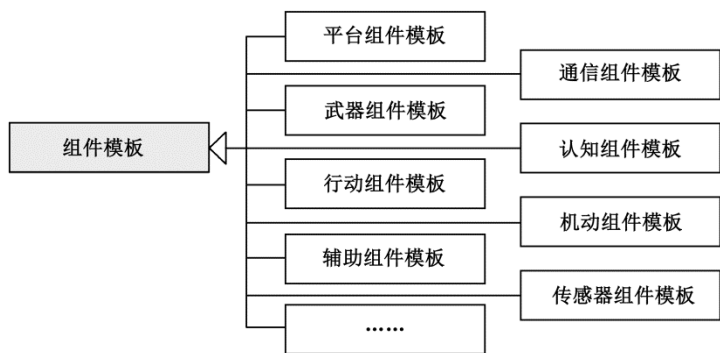


图 4-4 组件模板分类结构

平台组件模板主要是模拟作战实体的基本工作流程,约束作战实体的物理组成结构,规定实体可具备的目标属性^[41]。其型号化后的平台组件模型是组成实体模型的基本载体,平台组件模板的类视图如图 4-5 所示。

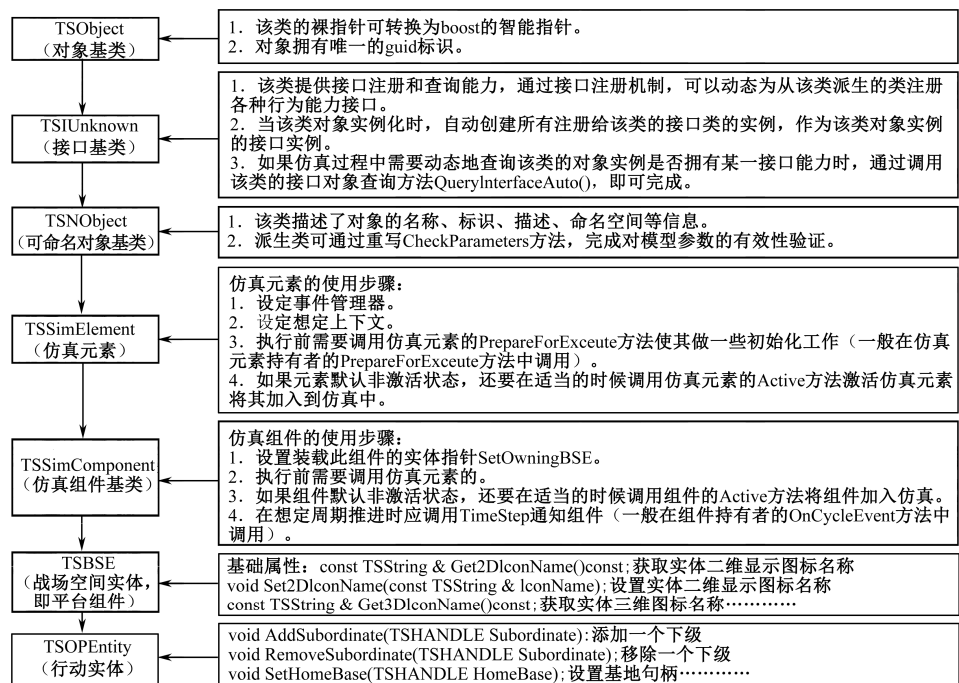


图 4-5 平台组件模板的类视图

通过对各种设备的固有物理属性和战技性能的模拟实现对其功能和作战能

力的仿真,图 4-6 和图 4-7 分别给出了机动组件模板和传感器组件模板的类视图。

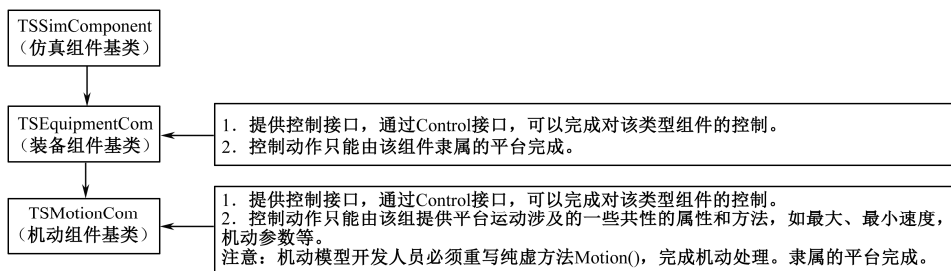


图 4-6 机动组件模板的类视图

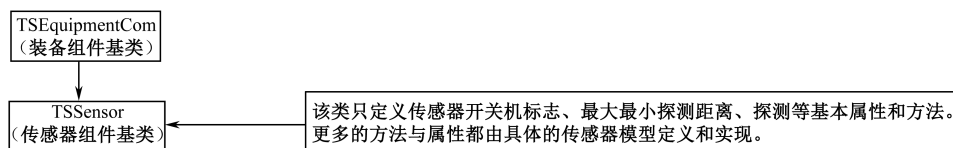


图 4-7 传感器组件模板的类视图

行动组件模板主要是对作战实体的战术任务、战术规则、决策准则等进行建模，模拟其在作战过程中的认知行为，图 4-8 给出了行动组件模板的类视图。

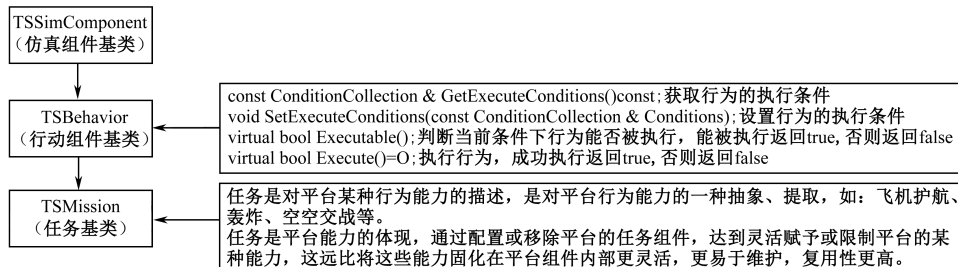


图 4-8 行动组件模板的类视图

辅助组件模板分为目标特征组件模板、交战裁决组件模板和网络组件模板等。目标特征组件模板用于模拟实体的 RCS、红外特征、外形特征等外部特征；交战裁决组件模板主要模拟裁决不同的毁伤组件对不同类型的目标造成的毁伤程度等^[42]；网络组件模板对连接作战实体的通信网络及通信使用的协议进行建模，模拟通信网络的基本特性，如网络类型、通信容量、通信速率等。图 4-9 给出了目标特性组件模板的类视图。

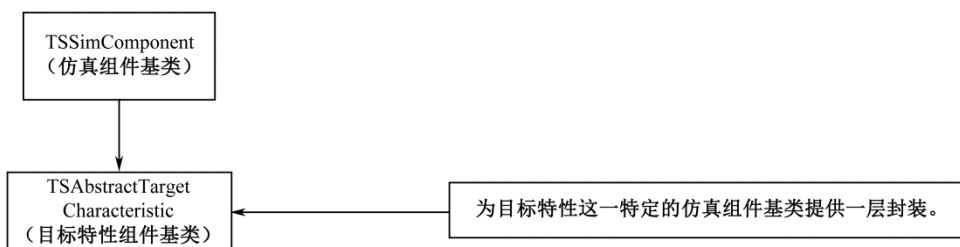


图 4-9 目标特性组件模板的类视图

2. 组件模型

实体模型对战场空间中的作战实体进行建模,通过对其物理原理、运行流程、行为准则的模拟,仿真其在战场环境中的作战能力和作战流程。实体模型的建模对象可以是单武器平台,也可为营、连级作战单位。为提高仿真模型的重用性、满足仿真需求的多样性,系统的建模粒度设定为组成作战实体的组件,由各类组件模型组合构成实体模型。

以飞机模型为例,采用组件的方法,不是对整个飞机建模,而是通过将不同的组件连接起来建模飞机,这将给建模人员提供更多的灵活性和重用的可能性。灵活性表现为通过采用不同的实例化机动组件和装备组件等来完成对不同型号飞机的建模;重用性表现为在多个平台上可以使用相同的传感器或武器组件。飞机模型组件化建模如图 4-10 所示。

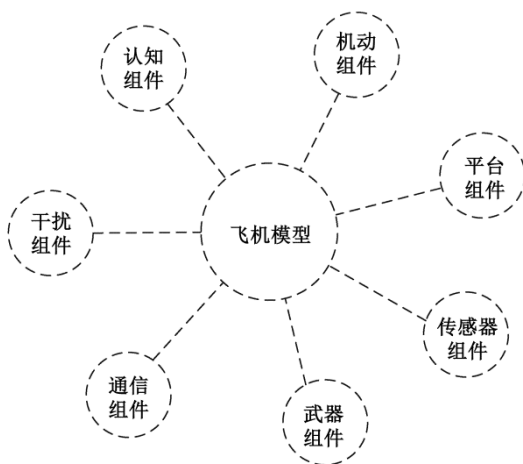


图 4-10 飞机模型组件化建模示意图

一个组件模板的建模对象是具有相同功能、原理、运行流程的物理对象集合。通过对组件模板的型号化,即设置组件模板的属性参数或配置行为流程,可生成不同的组件模型,如炮兵平台组件模板可型号化为122自行火炮模型、155自行加榴炮模型等。同一装备型号的组件在型号化过程中,根据仿真需求和基础数据情况可生成不同分辨率的组件模型。根据组件模板的分类结构,组件模型相应的包括平台组件模型、机动组件模型、武器组件模型、行动组件模型、认知组件模型、通信组件模型和辅助组件模型等,其分类结构如图4-11所示。

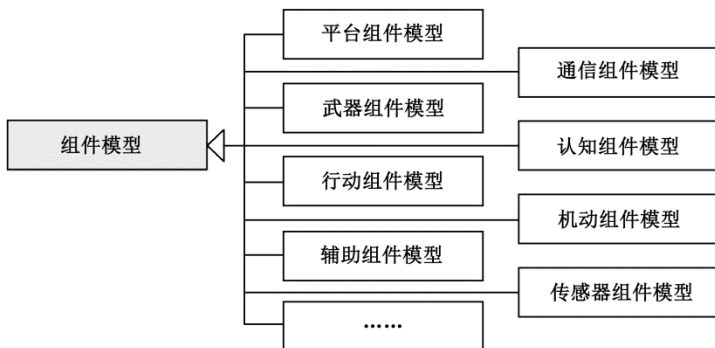


图 4-11 组件模型分类结构

4.2.4 组件开发流程

组件要求在设计过程中能够重用和组合,而且在运行过程中要求能够在仿真系统中即插即用。组件间只关注自身接口中需要外部环境提供的信息和自身能够为外部环境提供的信息,这种机制使组件功能的更新或变化不会影响到其他组件的变化^[43]。

组件开发包括平台组件和其他组件,两者的开发步骤大致相同,下面以平台组件为例介绍组件的具体开发流程,如图4-12所示。

平台组件的开发主要经过以下几个环节:①新建模型项目,并明确项目类型和名称;②定义不同平台、不同开发环境下模型的导出库;③建立模型类,并完成相关函数功能和消息处理功能;④完成动态链接库DLL的相关配置文件,加载模型并完成模型注册工作;⑤编写配置文件CC.mconf,打包后发布模型。

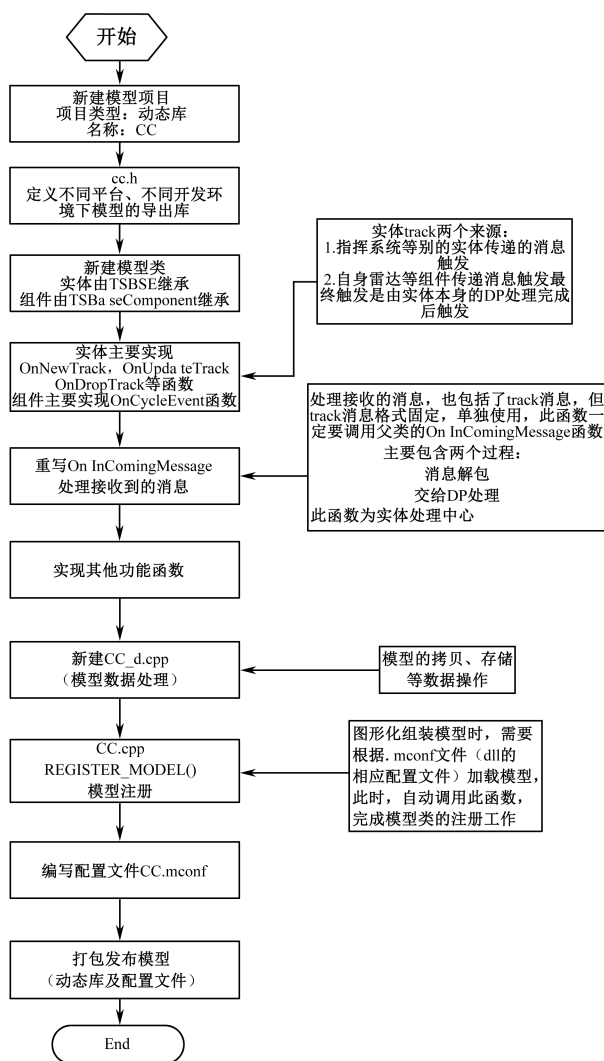


图 4-12 平台组件开发流程

模型接口是仿真模型组件的重要组成部分，良好的接口定义更是实现仿真模型组件组合与重用的关键。通过模型接口，组件模型之间才能进行数据传输，实现互联互通，完成仿真任务。只有符合仿真模型语法层接口规范的仿真模型才能在仿真系统框架中被调用和执行，组件相互之间才能协调运作。由于仿真模型一般是分布式的，因此，要求仿真模型组件还必须具有一致性的语义接口，组件之间的互操作需要标准的通信协议支持。整个模型开发流程如图 4-13 所示。

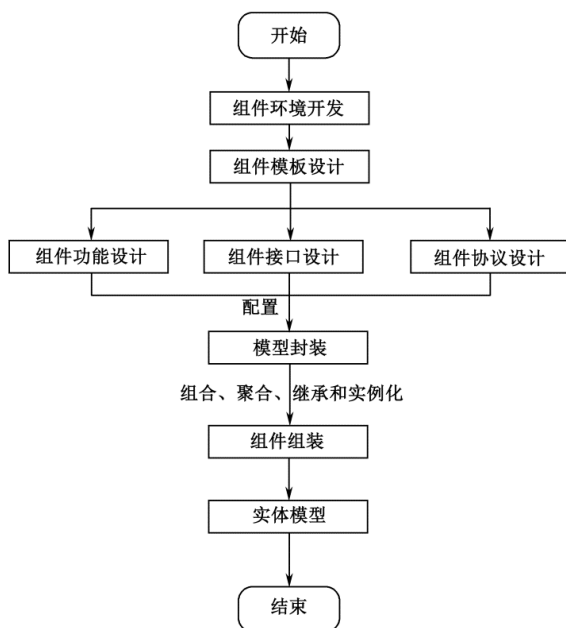


图 4-13 开发流程

组件到模型的开发主要经过以下步骤：①构建组件开发环境，并明确组件类型和名称；②根据模型类别设计组件模板；③以组件模板为基础，分别完成组件功能设计、组件接口设计和组件通信协议设计；④通过相应的配置后完成对组件的封装；⑤采用组合、聚合、继承和实例化的方式将多个组件组装成实体模型。

4.2.5 模型组件架构

模型组件架构主要由组件库、对象库、基本库和单元库构成，这四个库并非独立存在，而是相互关联，具有一定的映射关系、关联关系、组合关系和聚合关系，具体架构如图 4-14 所示。

(1) 组件库是原子库，按照组件化的思想，将每一个战场实体分割成多个不同的组件进行建模，所有的组件存储于此，包括认知组件、环境组件、平台组件、机动组件、武器组件、传感器组件、行动组件、通信组件、电子对抗组件和辅助组件等。单个组件不具备模型功能，任何一个实体需要有多个组件才能具备模型的能力。

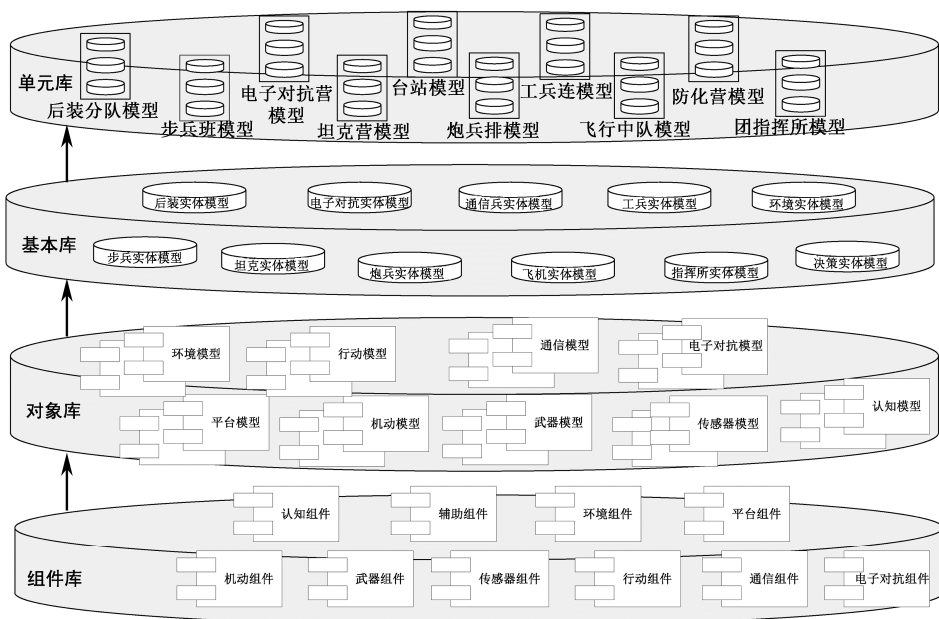


图 4-14 模型组件架构图

(2) 对象库是组件库中每个组件实例化的结果集。对每个组件模型的参数赋值后即生成一个组件对象模型，同一个组件在进行实例化时，会因赋予的参数不同而形成不同的对象模型。以机动组件为例，当机动速度为 60km/h 时，可以建模为坦克的机动组件；而当机动速度是 1500km/h 时，则可建模为飞机的机动组件。

(3) 基本库是由对象库按照军事概念空间的物理组合形式，将多种不同的对象模型按照一定的方式组合而形成的各类战场实体。基本库是对象库组合的结果，作为单个实体模型，基本库中的实体模型具备了独立的功能，能够直接完成简单的仿真活动。

(4) 作战单元库是基本库的一种聚合形式，主要是在基本库上的聚合级模型，是将多种类实体模型聚合、组装后形成的有独立作战能力的作战单元集合。根据建模仿真的不同分辨率要求，单元库存放了不同分辨率的模型，如炮兵排模型、炮兵连模型、炮兵营模型等。在单元组装时以平台组件为基础，将平台所搭载的各类传感器、武器、通信设备集成到该平台上形成在想定中存在的一个个仿真实体，一般具有机动、探测、打击、通信等一系列作战能力。

所有的仿真对象的建模结果都是组件，组件共分为以下十个大类。

(1) 认知组件主要是对指挥员的认知和决策行为进行描述，使各种实体有能力存储和处理信息，做出决策，从而实现有效指挥。认知可分为主动认知和被动认知。主动认知主要包括智能体组件和代理组件，根据相应的规则自行完成分析、决策和指挥；被动认知需要部分依靠人的认知完成一系列指挥决策活动^[44]。认知组件的公共属性通常包括认知能力、方式、主体、客体等。指挥控制组件主要对指挥能力、指挥层级等属性与指令产生、指令传输、指挥获取、指令处理等操作描述。决策组件主要对决策能力、决策层级等属性与任务感知、方案分析和决心下达等操作描述，如图 4-15 所示。

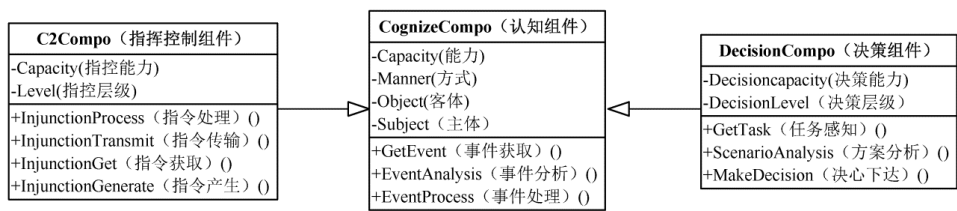


图 4-15 认知组件

(2) 环境组件主要是描述与作战行动、指挥活动相关的地理环境、战场环境、气象水文环境、电磁环境等战场环境因素的一类模型。该类模型应能与作战行动模型进行交互，模型描述的重点应该在于战场环境对实体行为的影响，如图 4-16 所示。

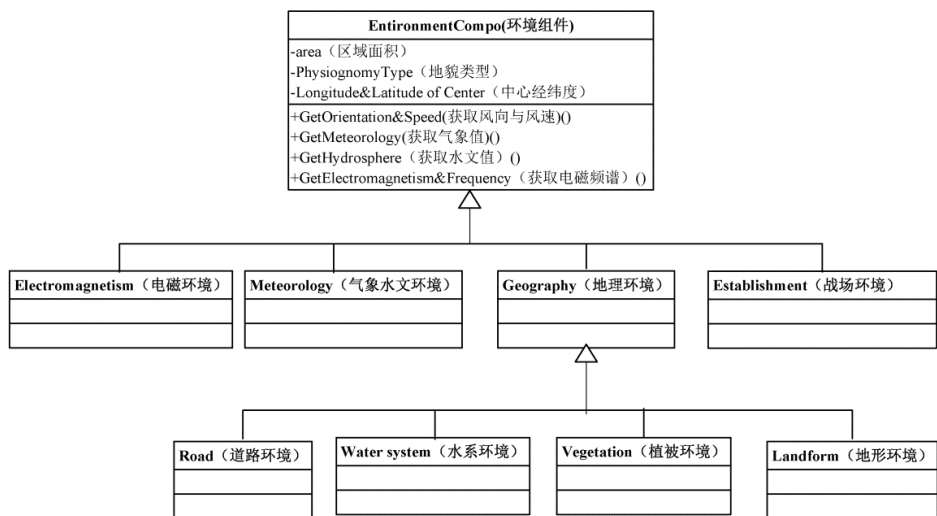


图 4-16 环境组件

(3) 平台组件通常是其他设备（如武器、通信等设备）的载体，任何作战实体都需要一个平台作为支撑，分为火力平台、通信平台、通用平台、指挥平台。平台是承载实体的基础，只有将不同的组件装配在平台上才能成为一个完整的实体，才能具备模型的基本能力，如图 4-17 所示。

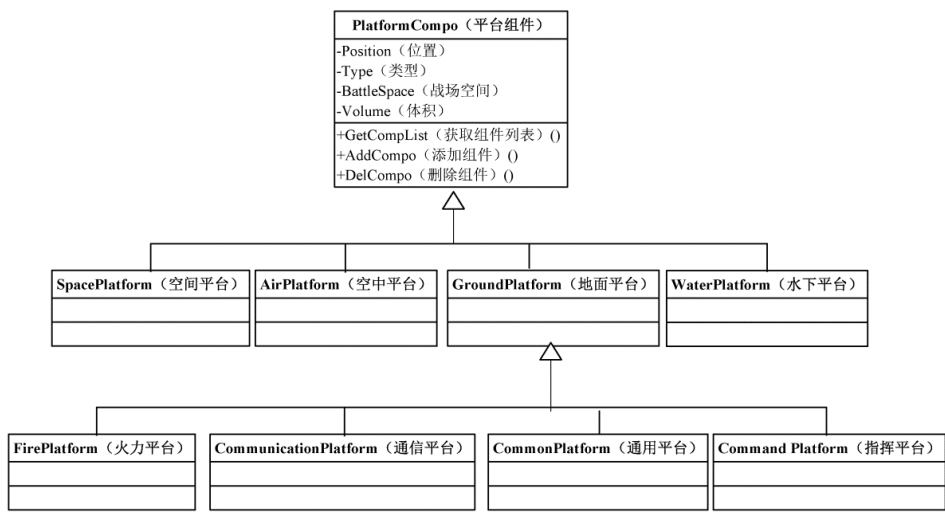


图 4-17 平台组件

(4) 机动组件主要描述实体在战场环境下的运动过程及机动设备的物理特性，针对平台的运动进行建模，如机身的运动、车辆的运动等。主要包括基于关键点运动组件模型、基于编队运动组件模型、基于道路信息运动组件模型、车辆避让组件模型等，通常与平台组件结合使用。地面机动分为轮式机动、履带式机动和轨道式机动等，如图 4-18 所示。

(5) 武器组件主要描述实体平台上搭载的地对地直瞄、地对地间瞄、地对空直瞄、空对地导弹等主战武器系统的装备管理、发射、运行和控制等，具体包括武器系统控制指令处理模型、火控系统仿真模型、弹药发射模型等。以弹药发射模型为例，主要是模拟间瞄武器发射的炮弹、导弹、火箭弹等主战弹药的性能及运动、制导过程，包括弹药运动学模型、飞行控制模型、发射指令处理模型、引信控制模型、弹道误差计算模型等，如图 4-19 所示。

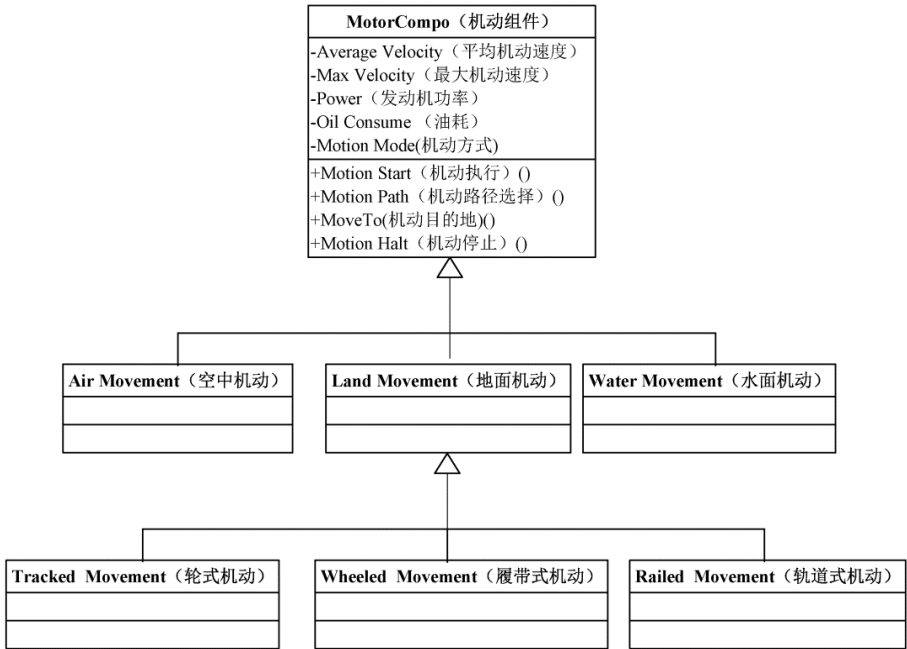


图 4-18 地面机动组件

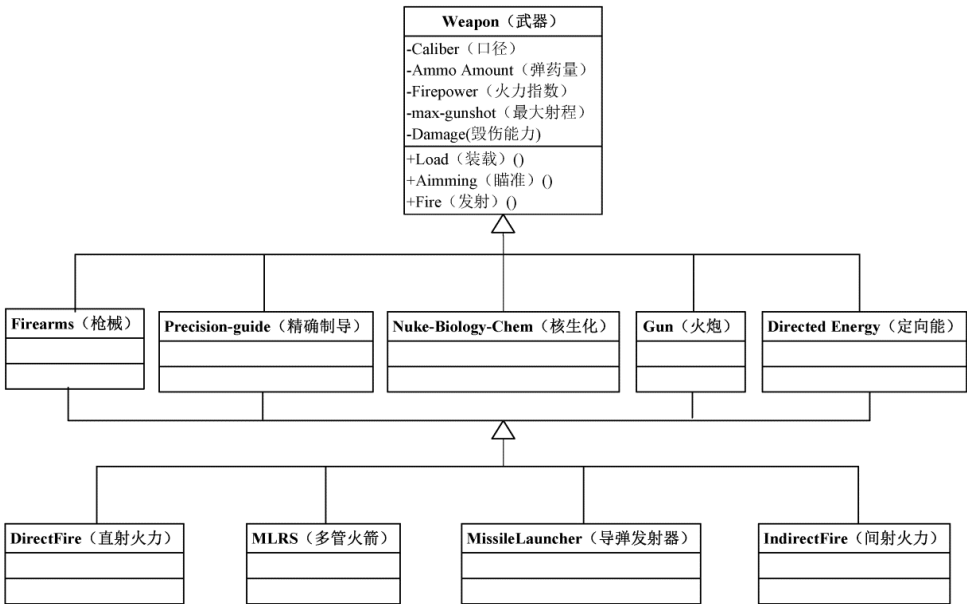


图 4-19 武器组件

(6) 传感器组件主要描述光学传感器、对空监视和搜索雷达等探测设备在考虑地形天候条件下的探测、侦察过程。传感器既可以对实体进行探测,也能够对加载在实体上的装备固有的物理特性进行探测。具体包括雷达传感器模型、红外传感器模型、光学传感器模型、无线传感器和网络传感器模型等,如图 4-20 所示。

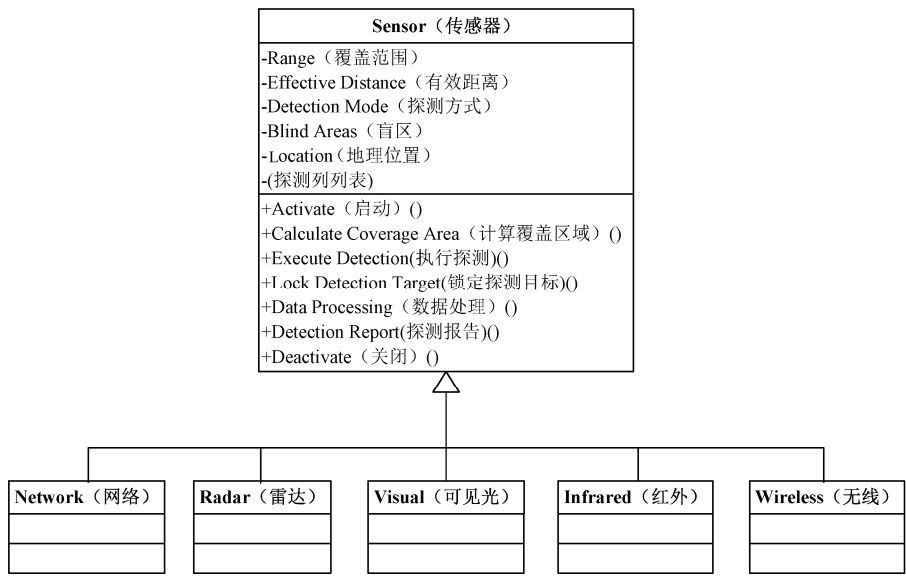


图 4-20 传感器组件

(7) 行动组件描述作战单元的基本作战动作,以行动实体为基本描述对象,主要对行动实体的属性、行动类型、在各类行动中可能实施的战斗动作及各类战斗动作的实施规则进行描述,包含机动、调整火力、攻击、撤退、补给等。每个作战单元的作战行为实际上是一个基本作战动作序列,基于这些动作序列可以完成各种作战任务。行动组件按照作战方式和阶段的不同,可以分为侦察行动组件、火力打击组件、机降组件、集结组件和攻击组件等,如图 4-21 所示。

(8) 通信组件主要描述不同种类的通信设备的工作性能、运行状态、信息传输和处理等,由通信网络组件及通信协议组件两部分构成。通信组件主要用来进行不同实体之间的通信,可以是两点之间的,也可以是多点之间,还可以是以组网的方式进行通信。通信组件具有外部通信和内部通信的能力,其主要内容包括通信链路监测、消息发送、消息接收和消息处理等环节。按照其通信方式的不同,

通信组件可以分为计算机网络通信、光纤通信、卫星通信、接力通信和无线通信等不同的组件，如图 4-22 所示。

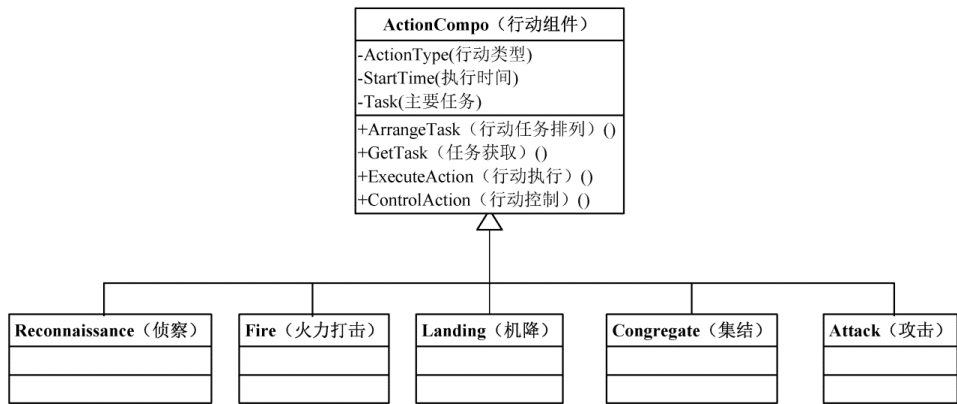


图 4-21 行动组件

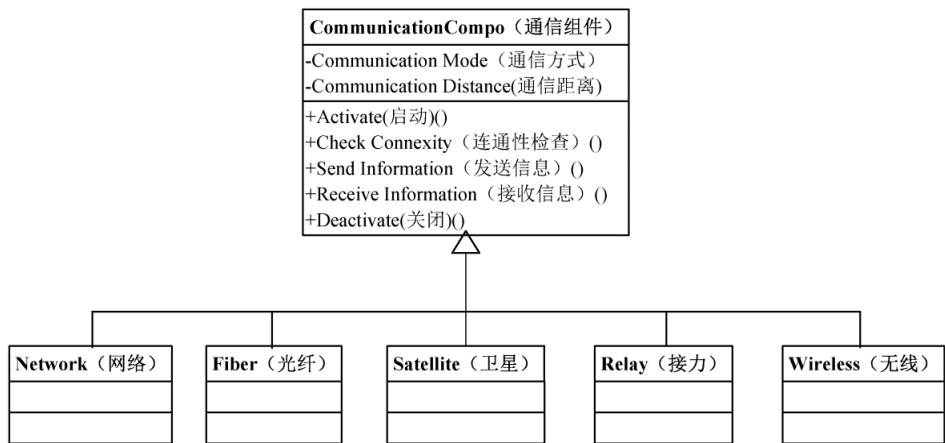


图 4-22 通信组件

(9) 电子对抗组件主要描述电子干扰、雷达干扰、通信干扰等多类干扰设备在考虑地形条件下的对抗干扰过程，目的是对对方的信号接收、信息获取进行不同程度的干扰。具体分为雷达对抗、通信对抗、光电对抗和电磁欺骗对抗等组件，如图 4-23 所示。

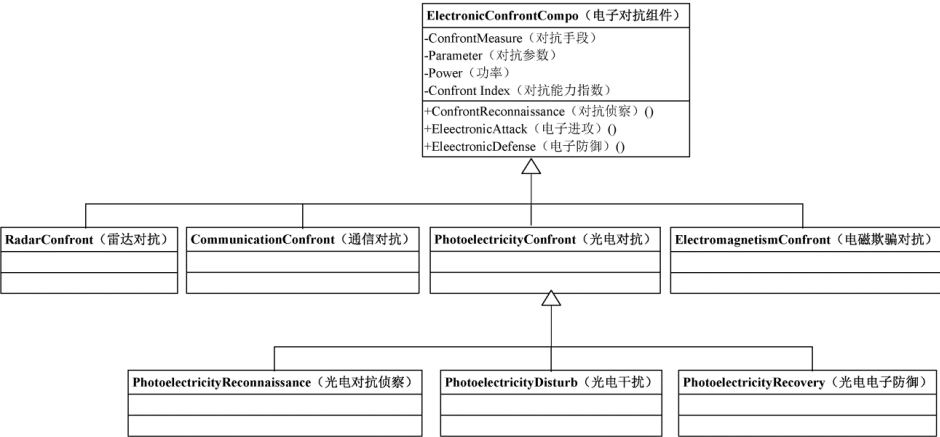


图 4-23 电子对抗组件

(10) 辅助组件主要是对上述所列组件之外的一些组件的描述，完成一些辅助仿真的功能，主要包括目标属性组件、作战属性组件、模型管理组件、火力毁伤效能评估和作战效果评估组件等。触发器组件如图 4-24 所示。

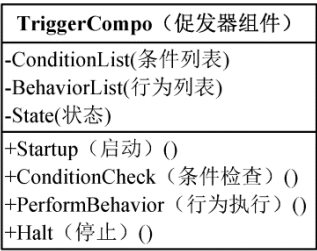


图 4-24 触发器组件

4.2.6 组件接口设计

组件接口描述了组件对外提供的服务，组件和组件之间需要通过接口进行交互。遵循统一的接口标准是组件实现重用和高度互操作性的基础，只有使用同样的接口标准才能保证组件之间可以互相通信、协同工作，共同完成仿真任务^[45]。在接口设计时先抽取系统内核心作战单元模型的共有属性和通用操作，完成通用接口设计；之后再以通用接口为基础，综合考虑组件扩展性能，按照接口准则定义组件特有的输入、输出接口。为提高模型组件的通用性，将模型组件接口设计分为组件描述接口、组件组合接口、组件功能接口、输入和输出接口等，具体如图 4-25 所示。

(1) 组件组合接口是实现与其他组件之间的组合调用，组件模型采用参数和功能分离的设计思路，在方便封装的同时能够实现服务之间的访问。

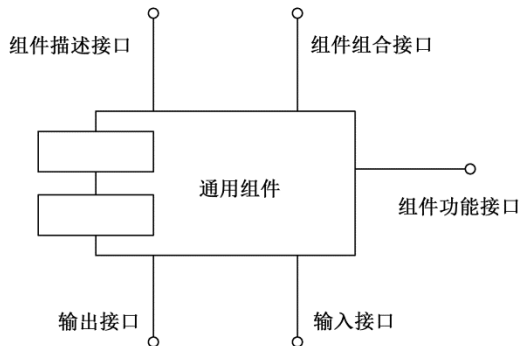


图 4-25 模型组件接口设计

(2) 组件描述接口用于描述该组件的功能和接口信息，为组件组装和管理提供支撑，其中包括组件名称、组件 ID、组件功能说明和组件接口说明等。

(3) 组件功能接口是系统预定义的功能函数调用接口，用于仿真引擎控制调用组件功能。

(4) 输入接口是组件实现功能所需的信息输入接口，按交换的信息内容不同可分为状态信息接口和事件信息接口两大类。状态信息接口包括目标状态信息等。事件信息接口包括装备控制信息和动作执行输入信息等。

(5) 输出接口是组件对外输出的信息接口，其内容与该组件的具体功能相关，该接口也分为状态和事件信息接口两类。

4.2.7 组件装配

组件装配的基本原理如图 4-26 所示，图中组件 A、组件 B 和组件 C 是 3 个独立组件，为了实现对一个更为复杂模型的建模，需要通过对组件 A、B 和 C 装配组合后得到新的组件 D。组件之间根据建模需要，按照接口对应关系建立接口间的连接，各组件模型在事件和消息的共同作用下完成新的组件功能。组件 D 组装完成后，只需输入相应的参数就能直接获取输出结果。由于本书中采用面向服务的方式管理组件模型，因此，每个组件都是以服务的形式存在，并以服务的形式对外提供功能，在后面的章节进行详细的分析。

组件化建模的核心内容是组件的开发和组装。组件本身是孤立的个体，一般无法独立实现作战单元的具体任务，只有将多个组件组装在一起并由仿真引擎组

织调用后才能完成指定任务。组件组装要解决两个问题：①如何以图形方式建立友好界面，使用户可以对组件进行管理，包括组件生成、修改、复制和删除等。②如何将组件进行组装，形成一个有独立作战能力的作战单元。组件组装管理结构如图 4-27 所示。

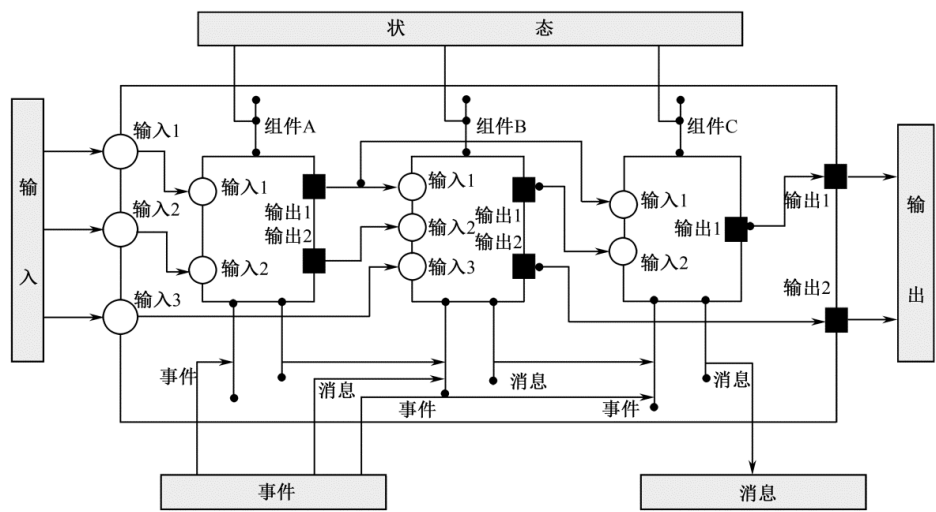


图 4-26 组件装配基本原理

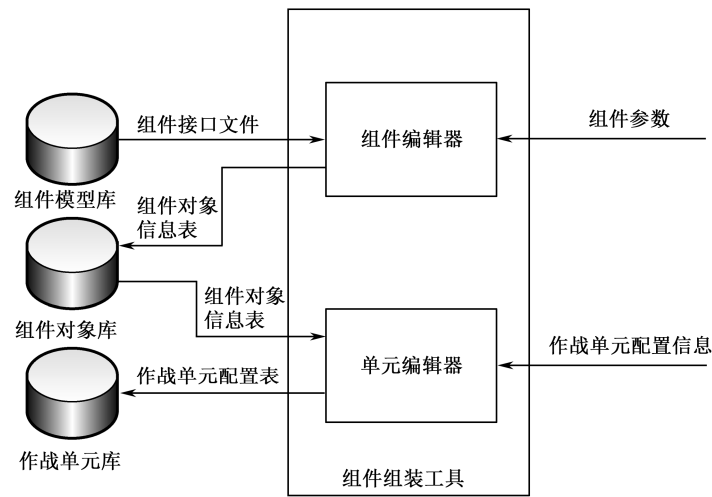


图 4-27 组件组装管理结构设计

组件组装工具由组件编辑器、作战单元编辑器两部分构成。

(1) 组件编辑器，用于编辑生成模型组件对象，从而形成系统组件对象库。组件编辑器通过读取组件模型库中已有组件模型的组件接口文件，自动解析组件初始化接口信息和功能描述信息并在界面上显示组件对象性能参数的类型和名称，由用户输入参数值从而新建一个型号的组件对象并存储在组件对象库中。

(2) 作战单元编辑器，用于将多个组件对象组装，产生新的作战单元并存储在系统作战单元库中。作战单元编辑器可以读取选择组件对象库中已有的多个组件对象，并定义单元配置信息，从而生成所需的作战单元，存储在系统作战单元库中。因此，在作战单元库中每个作战单元实际上是由一套作战单元配置表唯一定义的。

以坦克实体模型为例，下面介绍其装配过程。对于坦克实体模型的建模，需要组合机动平台组件、传感器组件、通信组件、武器组件和联勤保障组件等，如图 4-28 所示。机动平台组件是具有运载武器和机动能力的组件；传感器组件对攻击对象目标及其他威胁具有探测能力，并且可以将感知的探测信息传送至武器组件，在火力射程内完成打击任务；通信组件主要是用于坦克指挥实体与上级指挥机构、保障分队及其友邻部队之间进行信息传输，如通信组件与联勤保障组件建立联系，并实时告知坦克油料、弹药消耗情况。

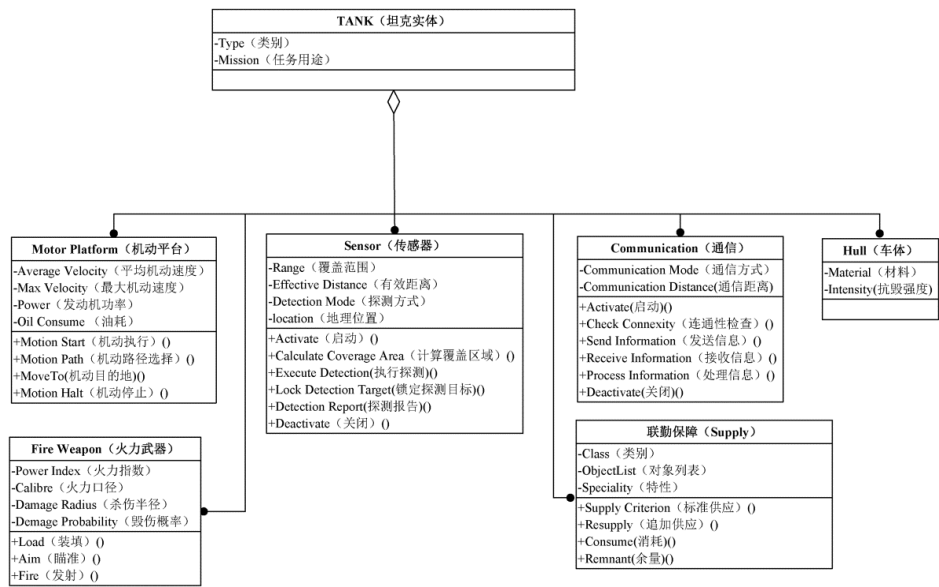


图 4-28 基于组件的坦克实体模型

4.3 模型通信及其消息管理机制

模型之间可以通过端口传递数据，子模型如果需要和外部模型通信，则必须通过组合模型。每个模型都有个模型基类，模型基类除了定义输入/输出端口用于传递事件外，还定义了管理子模型所需的属性和方法。通过模型基类，可以动态增加或减少子模型之间的输入/输出端口，也可以动态改变子模型之间的接口耦合关系。

模型之间的数据传递有两种方式：消息和过程调用。为了保证模型的运行效率，在组合模型内部，模型之间的数据传递采用过程调用实现，不需要通过仿真引擎进行事件规划。被调用的模型可以直接返回结果，不涉及时间同步问题，没有时间延迟。不在同一个组合模型内部的两个模型则通过发送消息实现数据传递。消息通过仿真引擎发送到目标模型并规划执行。采用消息的方式传递数据使得模型之间进行并行计算成为可能。

在仿真过程中，模型间通常需要进行交互与通信，在基于组件化的建模思想指导下，平台中的交互、通信本质上都是组件与组件之间的通信。图 4-29 给出了平台内部分组件模型之间的通信方法。

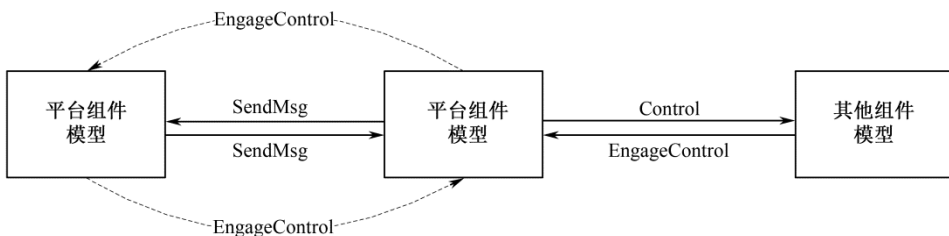


图 4-29 组件模型之间的通信方法示意图

图 4-30 给出了构成一个实体的各组件之间的一种数据通信方法。

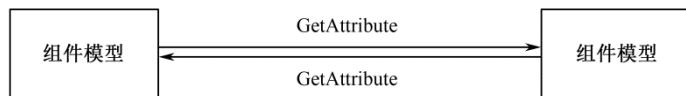


图 4-30 实体之间数据通信方法

图 4-31 给出了平台中 Track 信息数据交互示意图。

4.3.1 基于消息的通信方法

仿真实体之间的通信，其本质上是平台组件模型之间的通信，实体之间通过消息的发送与接收，完成指令及数据的共享，主要可以通过两种方式完成消息的传递：模拟网络和不模拟网络。

不模拟网络的方式是直接通过消息管理中心，完成消息的发送，采用 `SendMessage` 和 `ReceiveMessage` 方法完成。

模拟网络是基于平台自身的通信设备，将消息发送到既定的战术网络上，而后由网络模型完成消息的传输过程，最后由接收方的通信设备进行消息的接收。

下面以一个简单案例说明消息的创建、发送及接收处理过程，以此介绍实体间的通信原理。

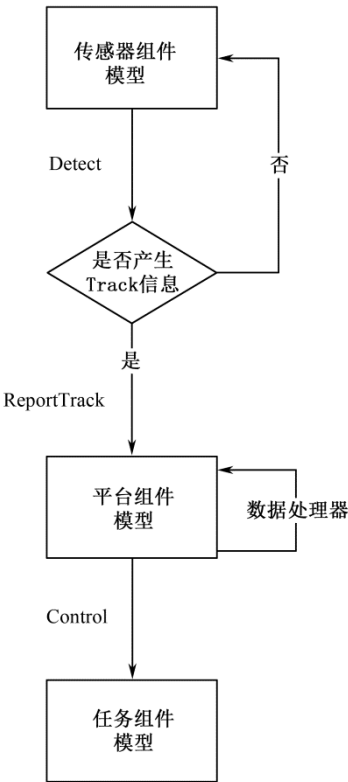


图 4-31 Track 信息数据交互示意图

消息类
- MessageID:UINT32
- Receiver:TSHANDLE
- Sender:TSHANDLE
- SendTime:TSTimeDuration
- Parameter:TSVariant
+SetMessageID():void
+GetMessageID():UINT32
+SetSendTime():void
+GetSendTime():TSTimeDuration
+SetReceiver():void
+GetReceiver():TSHANDLE
+SetSender():void
+GetSender():TSHANDLE
+SetParameter():void
+GetParameter():TSVariant

图 4-32 消息的类视图

基本背景：设指挥所向下级导弹发射单元指派救援战场受袭友军的任务，首先向指定单元发送消息（任务），当导弹发射单元接收到消息（任务）并确认是救援任务时，执行救援任务。对于此类消息的处理包含四个基本要素：消息的标识 `MessageID`，消息的发送者 `Sender`，消息的接收者 `Receiver`，消息的内容与参数 `Parameter`。消息的类视图如图 4-32 所示。

下面分析指挥所向下级导弹发射单元指派救援战场受袭友军的任务这一通信过程，过程代码如下：

```

TSMMessagePtr msgEnsure ( new TSMMessage( ) );
msgEnsure->SetSender(GetHandle());
msgEnsure->SetReceiver(eachHandle());
msgEnsure->SetMessageID(MSG_ENSURE);
msgEnsure->SetParameter(childHandle());
SendMsg(msgEnsure);

```

首先, 需在所对应的函数里 new 出一个消息类, 设置好消息的 ID、发送者、内容、接收者等信息后, 调用函数 SendMsg 即可完成消息的发送。导弹发射单元通过导弹发射单元平台组件类中提供的函数 OnInComingMessage 来对接收到的消息做进一步的处理, 示例代码如下:

```

Void HRSSFU:: OnInComingMessage (TSMMessagePtr Message)
{SuperType:: OnInComingMessage (Message);
//判断如果接收到的消息类型为支援任务, 则进行下一步的处理
If (Message->GetMessageID( )==MSG_ENSURE)
{.....
}
}

```

4.3.2 基于控制指令的通信方法

平台组件和其他组件之间的通信是通过发送控制指令的方式完成的。其中, 平台组件通过发送 Control 控制指令与其他组件进行通信; 其他组件通过消息管理器发送 EngageControl 指令记录当前状态, 平台组件能够对记录的状态进行感知, 从而达到与平台组件通信的目的。下面以一个简单案例详细说明各控制指令的创建、发送及接收处理过程, 以此说明平台组件与其他组件之间的通信原理。

基本背景: 设在想定过程中搜索车平台组件控制其机动组件向目标位置机动, 并在过程中开启携带的雷达, 对周围区域进行探测。若发现目标, 生成目标的跟踪信息, 并上报指挥所。搜索车平台组件控制其机动组件向目标位置机动这一过程通过搜索车平台组件向机动组件发送沿道路机动控制指令完成, 控制指令发送代码如下:

```

Void Control (UNIT32 Command);

```



```

Void Control (UNIT32 Command, const TSVariant & Parameter);
TSGenericMotionParamPtr Dparam= boost::make_shared<TSGenericMotionParam> ( );
Dparam->SetDesiredPos(PosPoint);
Dparam->SetCmdSpeed(CmdSpeed);
GetMotionCom( )->Control (C2_movevector,TSVariant::FromValue(Dparam));

```

这里, C2_movevector 是平台提供的朝指定点机动的控制指令, TSVariant::FromValue(DParam)为该指令的参数, 这里包含了机动的目标位置、指令速度等信息。机动组件通过自身的 OnControl 来对接收到的 Control 控制指令并做处理, 代码如下:

```

Void HRMotionCom:: OnControl (UNIT32 Command, const TSVariant & Param)
{
    SuperType:: OnControl (Command, Param);
    //判断如果接收到的消息类型为支援任务, 则进行下一步的处理
    If (Command ==C2_movewaypoint)
    {
        // 添加当接到此控制指令时需完成的工作
    }
}

```

雷达发现目标后, 搜索车上的数据处理器对目标进行跟踪, 通过消息管理器发送 EngageControl 指令完成, 代码如下:

```

TSRunContext::BattleMgr->EngageControl (Gethandle( ),C2_NEW_TRACK,Target->
Gethandle( ),TSVariant::FromValue(Track));

```

平台中函数 EngageControl 定义如下:

```

Virtual void EngageControl (TSHANDLE Acting,UNIT32 Action, TSHANDLE Against,
const TSVariant &Parameter=TSVariant( ))=0;

```

该函数主要用于 C2 指令的处理, 如数据采集、实体状态处理等。函数中各参数含义如下: Acting 表示交战控制指令记录方, Action 表示交战控制指令名称, Against 表示交战控制指令感知方, Parameter 表示交战控制指令的参数。

C2_NEW_TRACK 是自定义的切换声纳的控制指令, 该指令的记录方为搜索车携带的传感器, 感知方为目标实体, 指令参数为 Track 信息。搜索车实体可通过搜索车平台组件类中提供的函数 OnTruthEvent 来对感知到的 EngageControl

控制指令做处理，示例代码如下：

```
Void   HRDippingSonderEntity:: OnTruthEvent (TSHANDLE Acting,UNIT32 Action,
TSHANDLE Against, const TSVariant &Parameter)
{
    SuperType:: OnTruthEvent (Acting,Command, Against, Parameter);
    If (Command == C2_NEW_TRACK&& Acting==GetDataProcessor( )->GetHandle())
        {
            // 处理代码
        }
}
```

若目标实体能够感知这种态势，可用在其平台组件类中提供的函数 OnTruthEvent 来对感知到的 EngageControl 控制指令做处理，示例代码如下：

```
Void   OnTruthEvent (TSHANDLE Acting,UNIT32 Action, TSHANDLE Against, const
TSVariant &Parameter)
{
    SuperType:: OnTruthEvent (Acting,Command, Against, Parameter);
    If (Command == C2_NEW_TRACK&& Against==GetHandle())
        {
            // 处理代码
        }
}
```

4.4 作战仿真模型体系结构及内容

陆军仿真模型体系的建设内容，应能满足陆军部队在遂行多样化军事任务中多种作战行动仿真的需要。在本书中，主要针对陆军师、旅、团、营、连、排作战行动在不同作战层次的仿真需求来构建其相应的模型体系。

4.4.1 模型体系需求分析

陆军部队作战仿真模型应能够满足不同层次联合作战行动仿真需要,是由覆盖陆军部队作战指挥、作战行动及战场环境等系列模型构成的模型体系。这个体系,纵向覆盖联合战役、军种战役、合同战术(合成师、旅、团)和分队战术(营连、主战信息化武器平台)等多个层次,横向应包括陆军的合成、装甲兵、炮兵、防空兵、通信兵、防化兵等兵种(专业)。

作战仿真模型体系是整个仿真模型的军事框架,作战仿真模型体系要从其参与联合作战任务空间总体要求出发,对其参战实体、作战行动、指挥控制和作战评估等模型进行合理分类和科学管理^[46]。因此要求该模型体系,纵向应覆盖合同战术和分队战术等多个层次,即“纵向到顶”;横向应包括陆军部队诸兵种(专业),即“横向到边”。考虑到陆军部队作战仿真的需要,模型体系的“横向”主要界定在师、旅、团、营、连、排和班一级。从内容上讲,应该包括陆军作战行动中的武器装备模型、具体作战行动模型、各级指挥机构的指挥控制模型、交战模型、分析评估模型及作战行动过程中战场环境模型等。

从面向实体的角度来看,陆军部队作战行动模型体系应包括四个层次(联合战役层、军种战役层、合同战术层、分队战术层)的六个主要种类(武器装备类模型、作战行动类模型、指挥控制类模型、战场环境类模型、分析评估类模型、综合作用类模型)和多个兵种专业的模型。

4.4.2 模型体系框架设计

通过前面的分析可知,面向实体的模型体系还不能很好地满足当前作战仿真的需要。为了提高仿真模型体系的灵活性、可扩展性和可组合性,设计了一种柔性模型体系结构,如图4-33所示。

其构建的基本思路如下:为满足陆军部队的作战仿真需要,需要构建满足多个层次、多个级别、多个类型和多个功能的模型体系,即从模型类型上讲,应包括作战行动、指挥控制、综合作用、分析评估、武器装备和战场环境等模型;从

兵种专业上讲，作战仿真涉及炮兵、防兵、防化及合成兵种等，通常这些兵种根据所处的层级不同，拥有的兵力和武器装备数量的不同，又区分为师、旅、团、营、连、排和班不同级别；从作战层次上讲，由于陆军可能担负多样化军事任务，涉及联合战役、军种战役、合同战术和分队战术等多个作战层次。如果单从某一个维度去构建作战仿真模型体系，必然会使模型体系的全面性、合理性、关联性、及层次性不足。

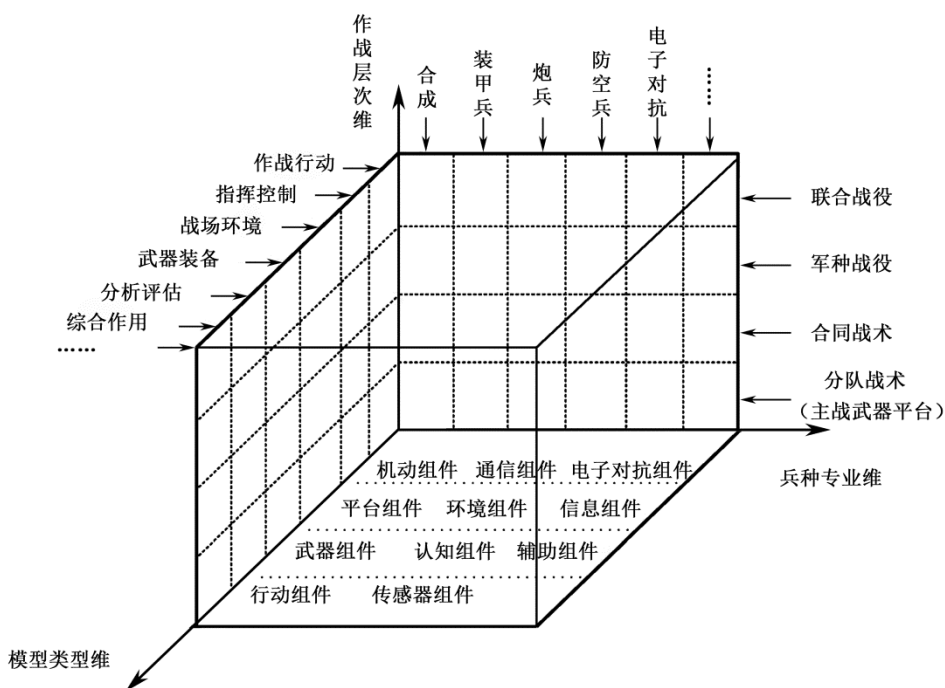


图 4-33 陆军作战柔性模型体系结构

为了解决上述存在的问题，设计一种“基础+维度”的可组合的柔性模型体系框架结构，即“基础”是指以各种模型组件（如机动组件、平台组件、通信组件、行动组件和辅助组件等）为基础，从模型类型、兵种专业和作战层次三个维度构建模型体系。

在此模型体系框架下，按照传统的模型建模方式，需要从这三个维度分别建立模型，会带来以下几个方面的问题^[47]：①增加建模的成本和工作量。由于模型的层次、类别和专业不同，在每一个维度上均要重新构建模型，且会增加很多重复工作。②从整体上看，从三个维度建立的模型之间相互独立，模型之间的关

系过于复杂,且这种复杂关系难以清晰描述。③模型的重用率低。从这三个维度建立的模型组合关系、继承关系、隶属关系和交互关系等很难描述清楚,对于公共组件也难以得到重用。

鉴于此,通过采用基于组件的方法构建作战行动、指挥控制、战场环境、武器装备、分析评估和综合作用等基础模型。上述基础模型可再划分为更具体的基础模型。以武器装备模型为例,可以划分为炮兵武器装备、防空武器装备及防化武器装备等,这里解决了模型类型维的构建。对于作战层次维和兵种专业维,则无须重新建模,而是在上述模型基础上,通过组合、继承(一个组件类具有另一个组件类所有属性和方法的能力,并可以增加其他属性和方法)、聚合(低分辨率模型向高分辨率模型聚合、下级单位或组织向上级组织或单位聚合)、解聚的方式得到。作战层次维主要是通过模型的组合、聚合方式完成联合战役、军种战役、合同战术和分队战术等层次的模型建模。兵种专业维主要是通过组件实例化过程、组合过程和继承等方式完成。不管是处在哪一个维度的模型,总是可以通过基础组件组合、装配后获得,然后再通过组合、聚合、组件实例化和属性配置等方式实现对其他维度的模型描述。因此,从逻辑上看,模型体系是以组件为基础,构建三种维度的模型,组成模型体系,便于建模人员清晰地理解和认识模型分类及模型之间的关系,这是解决体系问题的关键。从模型实际存在的状态上看,实际的模型存在于如图 4-33 所示的立方体中间,是三种维度的集合体。根据仿真应用的需要,可以按照对应的需求在不同维度上进行映射和分解,从而构建满足要求的仿真模型,解决模型的分类、管理和应用问题。

以炮兵营作战行动模型为例,按照传统的方式,必须采用直接面向实体的方式对此模型进行建模,而根据需求,如果要建立合同战术层炮兵团作战行动模型,则又需要重新建立模型。按照现有的模型体系,则无须进行重复的工作,直接可以通过对炮兵营模型的聚合,并对相应的人员、装备数量属性进行设置后得到,在增强建模灵活性的同时提高了建模效率。另外,随着建模仿真的发展,可能还会出现其他类型或层次的模型,也可以通过组件进行构建。对于不能满足要求的,可以通过增加组件完成。

4.4.3 模型体系内容

按照“基础+维度”的模型框架设计思路,构建了陆军作战仿真模型体系,

如图 4-34 所示。该模型体系以平台组件、机动组件、武器组件、行动组件、认知组件、传感器组件、通信组件和辅助组件等模型为基础，依据一定的逻辑关系、标准规范组装成各种实体模型。在构建具体的实体模型过程中，通过组合、聚合和继承等方式建立模型类型维、作战层次维和兵种专业维的模型。模型在实际仿真运用中并不是仅仅以某一个维度存在的，通常多维兼有。

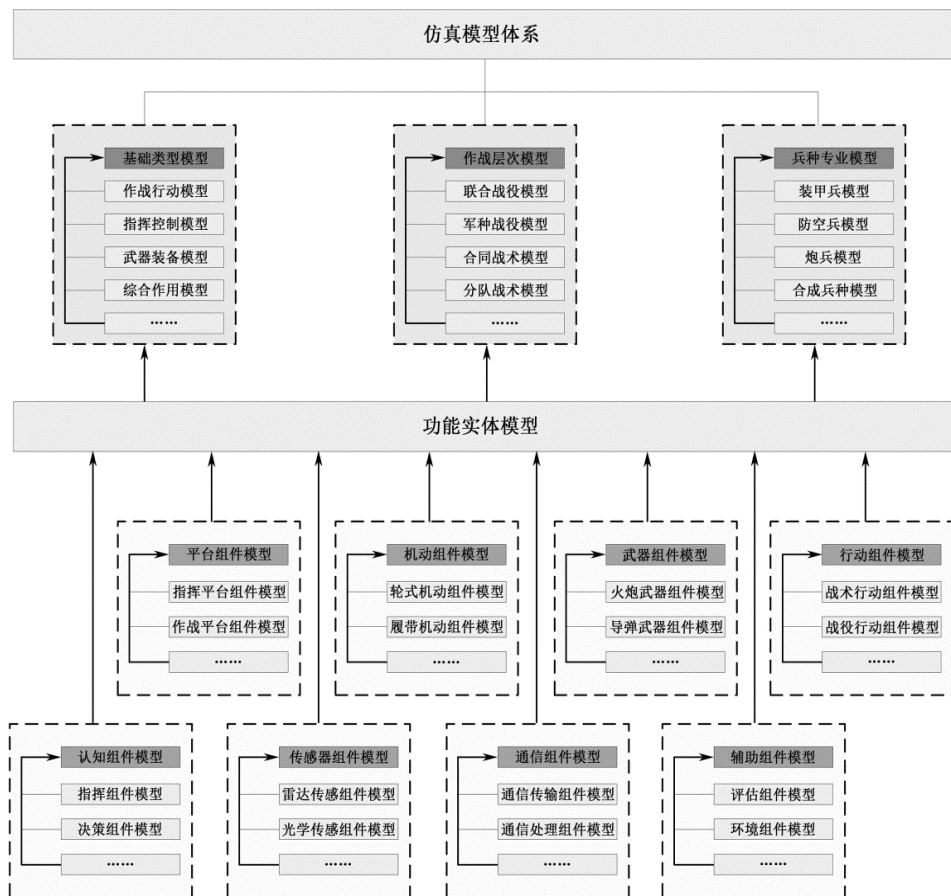


图 4-34 陆军作战仿真模型体系

为解决上述问题，将模型体系设计成以灵活、可扩展的方式展现，在逻辑上按照三个维度进行分类管理和区分，模型体系中并没有完全展开所有模型要素。以炮兵模型为例，在兵种专业维，分为炮兵排、炮兵连、炮兵营及炮兵其他保障

分队模型等；在模型类型维，分为炮兵进攻模型、射击模型、侦察模型和火力毁伤模型等；在作战层次维，分为炮兵单武器模型、分队模型、炮兵营指挥所模型、炮兵旅指挥所模型等等。下面分别对模型类型维度的基础模型进行介绍。

武器装备类模型以陆军部队现有的各类武器装备为描述对象，通过描述武器装备的战技术性能，对武器装备的作战运用过程进行建模，重点对武器装备的物理设备的属性和操作运用过程进行仿真建模，使其按照实装的方式完成相应的任务。

作战行动类模型以陆军部队作战中的行动实体为描述对象，对部队各类作战行动实体属性、作战行动过程、作战影响因素，以及行动实体与指挥实体之间、行动实体与其他作战行动实体之间、作战活动与作战环境之间相互作用的规则化军事定性和定量描述^[48]。着重描述实体的属性、任务类型、在各类任务中可能实施的战斗动作、规则，以及在实施各类战斗动作中，实体与战场环境间及与其他实体间的相互关系。

指挥控制类模型以各级指挥实体为描述对象，对各类指挥实体、指挥行为、指挥环境、影响因素，以及指挥实体与指挥实体之间、指挥实体与作战行动实体之间、指挥行为与指挥行为之间、指挥条件与指挥行为之间相互作用的规则化军事定性和定量描述。

战场环境类模型主要是描述与作战行动、指挥活动相关的地理环境、战场目标、气象水文、电磁环境、近地空间等战场环境因素的一类模型。该类模型应能与作战行动模型进行交互，模型描述的重点应该在于战场环境对实体行为的影响。

分析评估类模型主要是立足作战方案、作战计划模拟推演数据，按照方案论证和计划评估指标体系，对作战方案、作战计划进行专项或综合分析评估所需要的一类模型的总称。在仿真建模中分析评估类模型作为一类应用行为进行描述。

综合作用类模型主要是对各类战场综合作用现象进行描述的一类模型。该类模型主要用于描述战场各要素的综合作用，以反映多个作战实体综合形成的特殊作战效应，如兵力密度、信息共享、要素联动、分布指挥、精确保障等。综合作用类模型充分体现战场中基于信息系统的体系化作战的特点和规律。

4.5 本章小结

本章首先分析军事概念建模存在的问题,按照两个阶段、四个步骤的建模思路,在此基础上通过基于模板的概念模型描述方法,将概念建模区分为实体元模型、动作元模型、关系元模型、属性元模型和数据元模型。通过对上述元模型的建模完成对仿真对象的概念建模。重点对基于组件的模型设计方法进行了研究,设计了组件开发流程、仿真模型组件架构,以及可组合、可扩展的仿真模型体系,并通过对组件描述接口、组件组合接口、组件功能接口、输入和输出接口等完成对组件接口的描述与设计实现组件的装配,同时分别设计了两种模型组件的通信方式,最后完成了模型体系框架设计。

面向服务的作战仿真模型 组合方法



模型服务的目标是为作战仿真系统应用提供统一的军事模型规范, 不仅为模型的使用者提供一个共享存储与运行的平台, 而且也为模型的开发者提供标准规范。可以减少模型开发过程的无序性, 加快推进模型资源建设的质量, 提高军事仿真模型的可重用性^[49]。

服务具有组合的功能, 通过服务组合可以实现更为复杂的服务, 将每一个原子模型看成是一个服务。在执行组合模型时, 实际上就是执行多个原子模型的一种特定组合, 通过服务组合实现不同分辨率的模型组合。

5.1 模型组合的基本概念

模型组合是选择模型组件进行装配以满足不同仿真需求的一种能力。对模型进行组合首先必须有模型组件, 然后根据应用需求选择相应的组件进行装配。组件的装配是在某一软件平台上连接组件之间的输入/输出接口, 使组件之间能够按照要求传递数据。模型组合主要包括语法组合和语义组合两种方式。前者是关于如何实现模型组合的问题, 考虑的是如何在模型之间传递数据、模型之间的时

间同步及事件管理等工程实现问题；后者主要研究组合模型对于仿真系统是否有效的问题，即组合模型是否有意义。

对模型进行重构和组合的基础是组件技术，通过不同组件技术组合而成的模型可能无法运行，不同平台间的组件模型之间也可能无法相互调用。模型库系统一般包含多种类型的模型，且混合了许多与应用问题无关的仿真支撑框架信息。多个模型组合时粒度如何统一，每个模型的适用范围及集成的层次等问题，加大了模型集成的难度。对于不同层次的作战仿真而言，需要将多个不同层次的模型集成起来联合执行，重点就是要解决多模型间不同语义的理解和综合集成。

5.2 模型服务描述

各种组件或实体以服务的方式完成注册后，为了便于对服务的管理、调用和区分，需要对服务进行类型的划分和详细的描述。

服务的功能主要通过接口的描述和访问实现，服务之间的数据传递及交互控制都是通过输入接口与输出接口之间的参数匹配完成的。下面给出关于模型服务及服务组合的相关定义。

定义 5.1 服务 $S = \langle \text{Type}, \text{Name}, \text{Provider}, \text{Applicant}, \text{Inport}, \text{Output}, \text{Func} \rangle$ 为一个七元组。其中， S 表示服务； Name 是服务名称； Type 表示服务类型； Provider 表示服务的提供者； Applicant 表示服务的申请者，即获取服务的使用者； Inport 表示服务的输入接口和输入值集合； Output 表示服务的输出接口和输出值集合； Func 表示服务的功能，是服务的入口函数。

为了提升服务的自动组合能力，需要增强对服务的语义描述，主要包括两个方面，一是服务注册语义描述，这是进行服务查找的基础，服务描述的语义性越强，则查找到满足要求的服务的准确率越高^[50]；另一方面是服务请求语义描述，用户在服务请求过程中对模型要求的描述不可能十分专业，为了将用户的应用请求描述转化成具有一定语义的服务请求，使得用户需求与提供的模型能保持一致性，需要对服务请求进行形式化描述。

服务请求可以用一个常用的表达式进行描述： $SR_k(I_k, O_k)$ ， SR_k 是服务请求的名称， I_k 是服务请求的输入集合， O_k 是输出集合。

定义 5.2 服务语义关联 对于服务 $S_i(I_i, O_i)$ 和 $S_j(I_j, O_j)$ ，如果有 $O_i \supseteq I_j$ ，即服务 $S_i(I_i, O_i)$ 的输出集真包含服务 $S_j(I_j, O_j)$ 的输入集，则称服务 S_i 与 S_j 具有语义关联性，记作 $S_i \succ S_j$ ，其语义关联度表示为 $RS(S_i, S_j)$ ， S_i 称为前驱服务， S_j 称为后继服务。

定义 5.3 服务组合 为满足用户请求 SR_k 而构成的服务序列 S_1, S_2, \dots, S_n ，按照一定的原则组合在一起的服务集合，该服务序列需要满足下列条件：

- (1) $SR_k \succ S_1$ ；
- (2) 该序列中任意两个相邻的服务 S_i 和 S_{i+1} 都满足 $S_i \succ S_{i+1}$ ；
- (3) $S_n \succ SR_k$ 。

在组合过程中，模型被封装成服务。根据模型的复杂程度将服务划分为原子服务、基本服务、复杂服务和组合服务四种类型。

5.2.1 原子服务

原子服务对应于原子组件，是直接将单个组件模型直接封装形成的原子服务。这里的原子服务包括两种：一种是不具备独立功能的原子服务，如由机动组件直接封装而成的机动模型服务，这种模型单独使用，一般不具备行动能力，必须与其他组件或服务组合才具有仿真能力，如平台组件等；另一种是具有独立功能的原子服务，如辅助组件（机动路径规划组件、距离计算组件、射击诸元计算组件等）。这些组件直接封装后形成的原子服务，不需要依赖于其他的组件就能独立完成相应的任务。

5.2.2 基本服务

基本服务是指将原子组件按照一定的方式组装后形成具有一定作战行动能力的服务。以具备基本行动能力为判断标准，通常组成此类服务的原子组件不超过 10 个。以指挥所模型服务为例，指挥所实体=指挥所平台组件+地面机动组件+通信组件，通过组合封装后形成指挥所模型基本服务，其中指挥所平台组件、地面机动组件和通信组件是原子组件。

在服务组合过程中，基本服务是通过服务事件的形式进行描述的，且这种描

述具有唯一性，例如，TK 表示坦克模型基本服务，HP 表示火炮模型基本服务，如可用服务事件 E_TK 和 E_HP 分别标识 TK 和 HP 服务。对于基本服务和复杂服务，需要通过一个内部事件唯一标识它们的执行模块 INE，如可用 INE_TK 和 INE_if-!TK-then-HP 分别表示 HP 与 if-!HP-then-HP 的 INE 模块。基本服务拥有能够唯一标识其服务实例的 ID (IDentity)，ID 是服务的 URI (Uniform Resource Identifier)，服务事件和内部事件分别用以“E”和“INE”为前缀的字符串表示。NaDID 表示 Name 和 Dynamic ID，即系统命名和动态 ID，WSN 为服务名称变元。

式 (1) ~ 式 (7) 是对一个基本服务 ws 的 INE 模块描述，其描述过程包含如下：

步骤 1 当 INE_ws 发生但 ID 为 empty 时，规则 (1) 执行 getNaDID_ws 操作，并获得服务 ws 的系统命名和动态 ID，同时，规则 (2) 触发搜索服务 search_ws。

步骤 2 当处于搜索中且获得 ws 系统名称和动态 ID 的情况下，规则 (3) 执行 ac_Search 查找满足要求的服务实例，同时，规则 (4) 触发内部事件 conversation_ws 以使其进入会话状态。

步骤 3 当处于会话状态中且查找到合适的服务实例时，通过规则 (5) 执行 ac_Converse 以完成会话协议所要求的任务，同时，通过规则 (6) 使其进入应答状态。

步骤 4 当处于应答状态中且达成服务交易时，将该服务事件实例添加到服务事件序列中。

<i>INE_ws(ID) causes getNaDID_ws if empty(ID)</i>	(1)
<i>INE_ws(ID) triggers search_ws if empty(ID)</i>	(2)
<i>search_ws&echo_GND_ws(WSN,DID) causes ac_Search(WSN,DID)</i>	(3)
<i>search_ws&echo_GND_ws(WSN,DID) triggers conversation_ws(DID)</i>	(4)
<i>conversation_ws(DID) & echo_Search(ID,DID) causes ac_Converse(ID,DID)</i>	(5)
<i>conversation_ws(DID) & echo_Search(ID,DID) triggers answer_ws(ID,DID)</i>	(6)
<i>answer_ws(ID,DID)& echo_Conv(ID,AR,DID) attaches se_ws(ID,DID) if AR=approval</i>	(7)

5.2.3 复杂服务

复杂服务是针对基本服务而言的，是指具有多种作战行动能力的基本服务组合而成的服务。这种服务可能由两种方式组合而成，一种是以基本服务为基础对

象构建, 通过将多个基本服务组合形成, 具有多种行动能力^[51]。另一种是通过将多个作战行动组件按照一定的顺序组合, 形成一个完整的作战行动过程, 这种复杂服务可以提供对某一次战斗行动的仿真。对于不太关注每个行动细节的情况, 利用这种服务可以简化仿真过程, 提高模型运行效率。

构建过程构造子的目的是改善服务组合能力, 具体表现为以下两个方面: 一是通过构造子能够描述组合关系较为复杂的情况, 对于复杂的作战仿真模型而言比较适用; 二是通过构造子可以为常用的组合模型提供便利, 避免重复建模。这里主要构建三种过程构造子解决复杂模型的建模过程, 主要包括线序(Linear Sequence)、If-Then-Else 和 While-do。线序构造子用符号“;”表示, 通过线序构造子构建的服务组合顺序与执行顺序具有一致性^[52]。

设存在基本服务 s_1 和 s_2 , 式 (8) 和式 (9) 完成 $s_1; s_2$ 的 INE 模块描述。

$$\text{INE}_{s_1; s_2} \quad \text{triggers} \quad \text{INE}_{s_1} \quad (8)$$

$$\text{E}_{s_1}(\text{ID}_{s_1}) \quad \text{triggers} \quad \text{INE}_{s_2}(\text{ID}_{s_1}) \quad (9)$$

由判断条件的不同, If-Then-Else 构造子有两种不同的类型。一种是当判断条件是 $!s_1$ 时, 属于非此即彼类型, 采用 $\text{if-!}s_1\text{-then-}s_2$ 构建复杂服务; 而 $\text{if-C-then-}s_1\text{-else-}s_2$ 则是看固定的某种情况是否发生来确定不同的结果。两者的区别是: 前者将服务的执行状况作为判断条件, 也就是说如果服务 s_1 没有被执行则立即执行服务 s_2 , 而后者中则需对一个关键信息进行判断。如果关键信息为真, 则执行服务 s_1 , 否则执行式服务 s_2 。式 (10) 和式 (11) 是对 $\text{if-C-then-}s_1\text{-else-}s_2$ 的 INE 模块具体描述。

$$\text{INE}_{\text{if-C-then-}s_1\text{-else-}s_2}(\text{C}, \text{IM}) \quad \text{triggers} \quad \text{INE}_{s_1} \text{ if } \text{C}(\text{IM}) \quad (10)$$

$$\text{INE}_{\text{if-C-then-}s_1\text{-else-}s_2}(\text{C}, \text{IM}) \quad \text{triggers} \quad \text{INE}_{s_2} \text{ if } \neg \text{C}(\text{IM}) \quad (11)$$

While-do 构造子主要是针对同一服务被循环执行的情况, $\text{while-M-N-do-}s_1$ 描述表示 s_1 的同一实例被循环执行 $N-M+1$ 次。以炮兵射击为例, 为了计算第 M 发炮弹至第 N 发炮弹之间的毁伤效果, 需要射击模型循环执行 $N-M+1$ 次。式 (12)~式(14)是对 $\text{while-M-N-do-}s_1$ 的 INE 模块的具体描述。

$$\text{INE}_{\text{while-do-}s_1}(\text{M}, \text{N}) \quad \text{triggers} \quad \text{INE}_{s_1}(-1, -1, \text{M}) \quad (12)$$

$$\text{INE}_{\text{while-do-}s_1}(\text{M}, \text{N}) \& \text{E}_{s_1}(\text{ID}, \text{DID}, \text{CT}) \quad \text{triggers} \quad \text{INE}_{s_1}(\text{ID}, \text{DID}, \text{CT}+1) \text{ if } \text{CT} \leq \text{N} \quad (13)$$

$$\text{INE}_{\text{while-do-}s_1}(\text{M}, \text{N}) \& \text{E}_{s_1}(\text{ID}, \text{DID}, \text{CT}) \quad \text{attaches} \quad \text{E}_{\text{vs}}(\text{ID}, \text{DID}, \text{CT}-1) \text{ if } \text{CT} = \text{N}+1 \quad (14)$$

5.2.4 组合服务

自动服务组合主要需完成以下几个步骤：①服务的形式化描述，按照领域本体和描述逻辑的规范使得服务具有语义功能；②模型经组合后形成 WSDL 文档，按照服务描述的规范对其进行封装，并在 UDDI（Universal Description Discovery and Integration）中心进行服务的注册与发布；③按照用户的需求，自动找出满足要求的服务集合，并根据相关的算法进行选择、绑定，明确接口关系和执行顺序，同时进行服务流程化描述；④按照相关算法对发现和选择的服务进行自动组合、验证和执行。

服务组合的基础是进行模型服务发布，具体过程如图 5-1 所示。模型发布需要经过以下几个环节：①模型服务搜索。模型组件在封装注册前，需要通过 UDDI 查找是否存在相关或同名的服务组件服务。②组件服务分类命名。根据模型组件的功能为组件服务建立名称，并对模型组件服务的 WSDL 文档进行转换，同时产生基于 WSDL 的 UML 模型。③模型服务注册。在 WSDL 文档中提取模型语义信息，经过映射、转换后产生 BPEL4WS 代码，编译后形成规范的 WSDL 文档，在 UDDI 完成注册，并对系统用户进行发布。

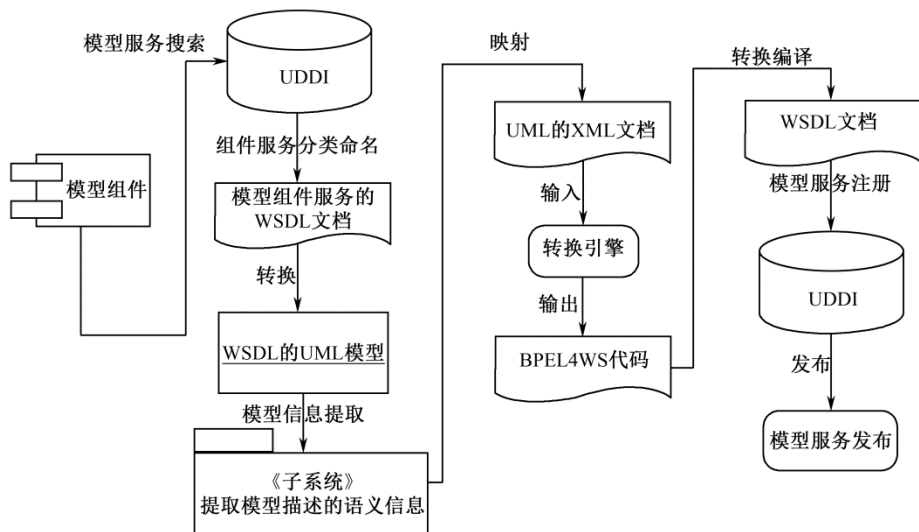


图 5-1 模型服务发布过程

5.3 面向服务的仿真模型组合建模方法

模型语法组合解决了模型组合的形式化问题,但是由于组合建模还依赖领域知识,例如,如何根据装备知识确定组合模型的子模型,如何表示组合模型的子模型必须满足的约束条件,如何根据领域知识确定组合模型的有效性,以及如何表达用户对组合模型的特殊需求等,目前的组合建模方法还不能很好地解决这些问题,为此,提出基于领域规则的组合建模方法,采用逻辑规则表示领域知识对组合模型的约束,建立了相应的模型组合规则和约束规则^[53]。同时,模型之间的接口匹配约束也通过逻辑规则表示,建立接口匹配规则。通过知识模板,把组合模型的接口连接数据和子模型数据转化为事实(Fact)知识,根据模型组合规则和约束规则,借助通用的逻辑推理引擎实现对组合模型的约束检查和接口匹配检查。事实上,在服务组合之前需要完成服务匹配,这里不作具体讨论。

5.3.1 组合建模需求

目前,组合建模技术能做到的是在一个共同的模型开发和运行环境下,通过定义一个用于组合的公共模型接口集合,在这个集合范围内实现模型组合,而且模型的选择和组合模型有效性判定等都还需要人工参与。由功能组件构建目标模型时,即使是相同类型的功能组件也有参数化的差异。此外,在模型组合过程中,组合模型受到很多领域知识的约束,表示模型行为语义及判断行为的等价性主要处于理论研究阶段,目前多数仿真系统的模型组合只能实现语法组合^[54]。语法组合只关心模型之间的接口匹配问题,这属于 I/O 观测层的模型重用。这种组合方式只要求描述模型的输入、输出接口。语法组合面临的主要问题是不能表示和处理与组合建模相关的领域知识,而这些领域知识对于组合建模是必需的。为了解决这个问题,采用逻辑表示模型和领域知识,通过逻辑推理统一处理模型接口匹配和领域知识对组合模型的约束,这种方式称为基于领域规则的模型组合方法。这个方法需要解决三个问题:①用知识表示方法描述模型;②用逻辑规则表示领域知识对组合模型的约束;③通过逻辑推理实现模型组合过程的检验。

5.3.2 服务组合方法

为简化模型组合的实现方式,同时又能满足多样化作战仿真对模型服务组合的需求,针对作战仿真模型组合规则复杂、基本服务类型相对单一等特点,提出一种基于事件的服务组合 BEWSC (Based-Events Web Services Composition) 方法。首先定义一种基于 MERA (Mission-Event-Restriction-Action) 的规则化语言,在此基础上,采用过程构造子按照模块化方法对模型服务的组合方案进行描述,解决了服务组合域表示困难、描述能力不足、模型匹配差异性大等方面的问题。

目前,服务组合方法主要分为基于流程、基于模型驱动和基于人工智能三种方法。基于流程的服务组合建模是一种基于过程模板的建模方法,其建模步骤为:先确定过程的各类成员,如活动、角色等,然后将它们定义成模板;再针对每个模板,在不同的抽象层次上构造该模板的具体对象;最后定义同一抽象层次中所有对象之间的各种关系,从而完成一个完整的过程模型^[55]。这种方法的优点是建模能力强,实用性好,同时也是增量式建模方法,可根据需要不断扩充模板的对象,而不会影响到整个过程模型的结构。

基于模型驱动的组合方法主要是在用特定语言定义组合服务的基础上,采用模型事件驱动的方法来开发、管理动态服务组合,模型驱动方法将软件开发方法学应用到服务组合中。该方法最大的特点是将组合逻辑与组合规范分离开,使用 UML 描述服务组合,使得能在更抽象的层次将服务组合模型化,该组合定义随后可自动地映射到特定的规范上,进而再关联到具体的服务上。

OWL-S 的出现为基于 AI 规划的组合方法提供了可能,将服务看成 AI 中的动作,通过输入/输出参数、前提和结果等来描述服务。在服务组合时,只要将服务的这些描述映射为动作形式化描述,在服务空间中以构造服务组合为目标,通过形式化的推理来得出服务的组合序列,动态形成服务组合方案,同时也能够保证规划结果的正确性和完整性。基于 AI 的服务组合方法能动态地生成服务组合方案,动态地链接具体实现的服务,能适应服务的动态变化。目前,如何高效、准确地在服务空间中查找适合的服务,如何评价生成的服务组合方案与需求吻合度等一系列问题还没有很好地解决。

为了提高模型的重用性,仿真模型体系中分辨率最高的模型单位是以组件的

形式存在的,因此,在仿真过程中,可以根据仿真应用的级别组装成不同的作战仿真实体模型,同时,为了满足作战仿真的需要,对于特定的仿真应用而言,单个实体的模型所提供的功能有限,难以满足大规模仿真的需求。例如,在一次作战行动中可能会涉及多个实体模型组合、协同等,有必要选择若干模型进行组合应用。在实际仿真过程中,模拟一次作战行动,可能需要多个实体按照时序完成多个行动。为了提高模型仿真运行效率,对于一些固化的作战行动而言,涉及的作战实体、军事规则和行动可以组合,经组合、封装后直接供指挥员使用,加快作战仿真进程;对于一些非固化的作战行动而言,涉及的作战实体、军事规则和行动可能是随机的,需要实时自动组合,这对模型的组合应用提出了更高的要求。采用基于服务的方式对模型进行管理和应用,可以简化很多流程,按照服务组合的方式可以实现更多、更复杂的仿真服务,以满足大规模、多任务的仿真活动。

本节提出了一种基于事件的服务组合方法。首先需要定义一种基于 EMRA (Event-Mission-Restriction-Action) 规则的语言,简称为 EMLCS (Event-Mission Language for Combined Services)。这里的任务可以理解在完成一次作战行动所发生的具体事件,即任务触发了事件的发生,一个事件的发生可能导致多个任务的产生。文献[55]提出了用于服务组合的 MCA 语言 EOVL (Evolving Logic Program),但其语言本身具有复杂性,组合效率不高。为了使组合过程中的语法更加简明,提高组合效率,在描述实时策略的 PDL (Policy Description Language) 基础上补充了 EMLCS 的定义。结合作战仿真的实际需求,主要在以下 3 个方面对 PDL 进行了完善:①对事件类型进行了扩充,在明确内部事件的同时,增加了服务事件;②增加了对服务组合规则详细描述;③补充了复杂事件的产生条件。

5.3.3 服务组合语法

EMLCS 中包含基本事件、任务、行动和常数 4 种相互互斥的符号集,其中,基本事件包括外部、内部和服务事件 3 种。外部事件的作用在于感知外部信息,如服务查询结果;事件触发后,一个事件被分解成多个任务,每一个任务又由若干个行动组成,行动的结果影响着外部世界。内部事件和服务事件的定义需根据组合方案而具体描述。内部事件的作用在于对服务执行状态等的描述,而服务事件则是对基本服务的一种标识。

event decomposes task according to military rule	(15)
task causes action if restriction	(16)
event triggers internal event if restriction	(17)
task produces a series of actions if rule	(18)
event attaches service event if restriction	(19)

EMLCS 中存在形如式 (15) ~ 式 (19) 的 5 种规则, 在解释这 5 种规则之前, 首先完成以下 4 个定义。

定义 5.4 事件 (event) 分为基本事件和复杂事件, 基本事件通过事件体 $E_i (1 \leq i \leq n)$ 来表征, 复杂事件表示成 $E_1 \& \cdots \& E_n$, $\&$ 为逻辑合取操作符。其中, 事件体分为正向事件体 $E(t_1, \cdots, t_n)$ 和反向事件体 $\neg E(t_1, \cdots, t_n)$ 。正向事件体表示在某个时间点事件 E 已经或正在发生, 逆向事件体则表示在某个时间点事件 E 还没有发生。事件发生是指正向事件的发生, 通过对变元的参数化获得事件实例。

定义 5.5 任务 (task) 在事件触发的条件下形成, 一次事件的触发可以产生并分解成多个时序任务 $T_i (1 \leq j \leq m)$, 每个任务可以由若干个行动 (action) 组成, 任务 $T(t_1, \cdots, t_n)$ 通过多个行动 $a_i (1 \leq i \leq n)$ 来表征, 可以表示为 $a_1 \& \cdots \& a_n$ 。

定义 5.6 行动 (action) 表示为 $A(t_1, \cdots, t_n) (n \geq 0)$, 其中, A 是行动名, t_1, \cdots, t_n 是它的 n 个变元, 变元 t_1, \cdots, t_n 必须出现在正向事件体中, 行动实例通过对变元的参数化获得。

定义 5.7 条件 (restriction) 是对事件、任务和行动顺利执行的规则约束, 不管是事件、任务还是行动都需在满足一定的作战条令、原则和规则条件下进行。它表示为 r_1, \cdots, r_n , 其中, $r_i (1 \leq i \leq n)$ 是表示比较关系的原子谓词, 逗号是一种逻辑合取。

满足条件时, 事件的发生将分别导致规则 (15) 中的任务按照军事规则和要求产生, 任务会根据规则 (16) 分解成具有一定顺序的子任务序列, 子任务会导致行动被执行, 触发规则 (17) 中的内部事件 (internal event) 和添加规则 (18) 中的行动序列, 最后将满足条件的服务事件 (service event) 添加到服务事件序列中。

5.3.4 服务组合规则

组合建模领域规则主要是实现领域知识对组合模型的约束, 通过规则表示实

现对接口匹配约束。根据约束的不同将领域规则分为五种类型：接口匹配规则、组装规则、约束规则、消息传递规则和效能预评估规则。

1. 接口匹配规则

接口是模型间的一种约定，也是一种模型解耦合机制，主要是提供对操作的访问。接口匹配规则是模型在组合过程中必须满足的约束条件，包括输入、输出、粒度和数据域条件等，模型接口匹配规则是进行模型组合的基础。

2. 组装规则

组装规则是指模型在组合过程中对模型类型的约束，根据作战行动过程要求，模型在组合时也必须按照一定的要求进行组装。模型组装规则总体上可以分为两大类：一是类型规则，即某一模型只能与其规定范围内的模型类型进行组合，主要是对被组合对象的类型要求；另一种是功能规则，主要是在实现具体功能时，模型组装的一些具体规则，即一个模型如果作为独立的模型参与仿真，它是不需要遵守这些约束的，但是如果它作为某个组合模型的子模型时就必须遵守这些约束。例如，通用飞机平台模型作为独立的模型参与仿真时，它的最大速度值可以取较大的值，但是当它作为电子战飞机的平台模型时，最大速度就要受到限制^[56]。

3. 约束规则

约束规则包括作战行动规则和条件判断规则。规则是模型的核心，任何模型的运行均是靠规则驱动的，作战行动规则是模型在模拟作战行动过程中需要遵守的规则，如作战条令和装备操作规范等。作战行动规则存在于模型运行的整个过程。条件判断规则是在模型遇到多种复杂情况下的判断依据，体现模型的多样性。在建模过程中需要多种规则支持才能完成相应的模型功能。

4. 消息传递规则

作战仿真中的模型交互通常通过消息的发送与接收进行，完成战术指令及情报数据的共享可以通过两种途径完成消息的传递：模拟网络或不模拟网络。模拟网络，平台通过自身的通信设备，将消息发送到既定的战术网络上，网络模型完成消息的传输过程，最后由接收方的通信设备进行消息的接收；不模拟网络，则直接通过信息管理中心，完成消息的发送。任何一种消息的传递都要按照固定的格式和协议进行，模型组合过程的消息传递则要根据作战仿真的需要选择不同的

消息传递方式。

5. 效能预评估规则

效能预评估的规则主要是对组合模型的综合效能进行评估。假设模型具有完备性,即用户所需的模型均能在模型服务中找到,根据用户描述的需求,在完成一次战斗行动过程中,没有限定具体的武器装备,但限制了能够提供的战斗力指数,因此在服务组合过程中,出现了由多种武器装备组成的不同组合方案,在这种情况下,应根据效能预评估规则,对不同组合方案的总体效能进行评估,选择效能较好的组合作为最优方案。

5.3.5 服务组合算法

服务组合的目的是将单个、独立的服务组合后去实现更复杂的服务,根据用户的仿真应用需求,通过组合的方式选择满足要求的服务,并按照一定的规则协同完成服务请求。实际上,对于仿真模型的组合问题相对于 Internet 中的服务组合要简单一些。对于同一种原子服务而言,Internet 环境中可能有多家服务厂商提供,因此,会出现多种组合方案,然后要根据一些最优化原则进行选择^[57]。而对于作战仿真模型而言,通常某一原子服务会由相应的权威单位提供,相对单一,不会出现太多的组合方案,更多的是关注原子服务的执行顺序和协同问题。

下面分别从服务的发布者、服务的请求者和推理过程这三个角度对组合服务的执行流程进行具体分析,如图 5-2 所示,三者执行步骤如下。

1. 模型服务发布者

Step1. 模型服务发布者结合作战仿真领域本体对自己发布模型的功能和信誉度以服务的方式进行语义描述,生成 OWL-S(Ontology Web Language based on Reputation for Services)文件。

Step2. 生成的 OWLR-S 文件提交给模型服务分类模块,服务分类模块根据发布服务的信息,对发布模型服务进行分类,便于用户查询。

Step3. 将 Step2 得出的分类信息写入 OWLR-S 文件,扩充和完善 OWLR-S 文件。

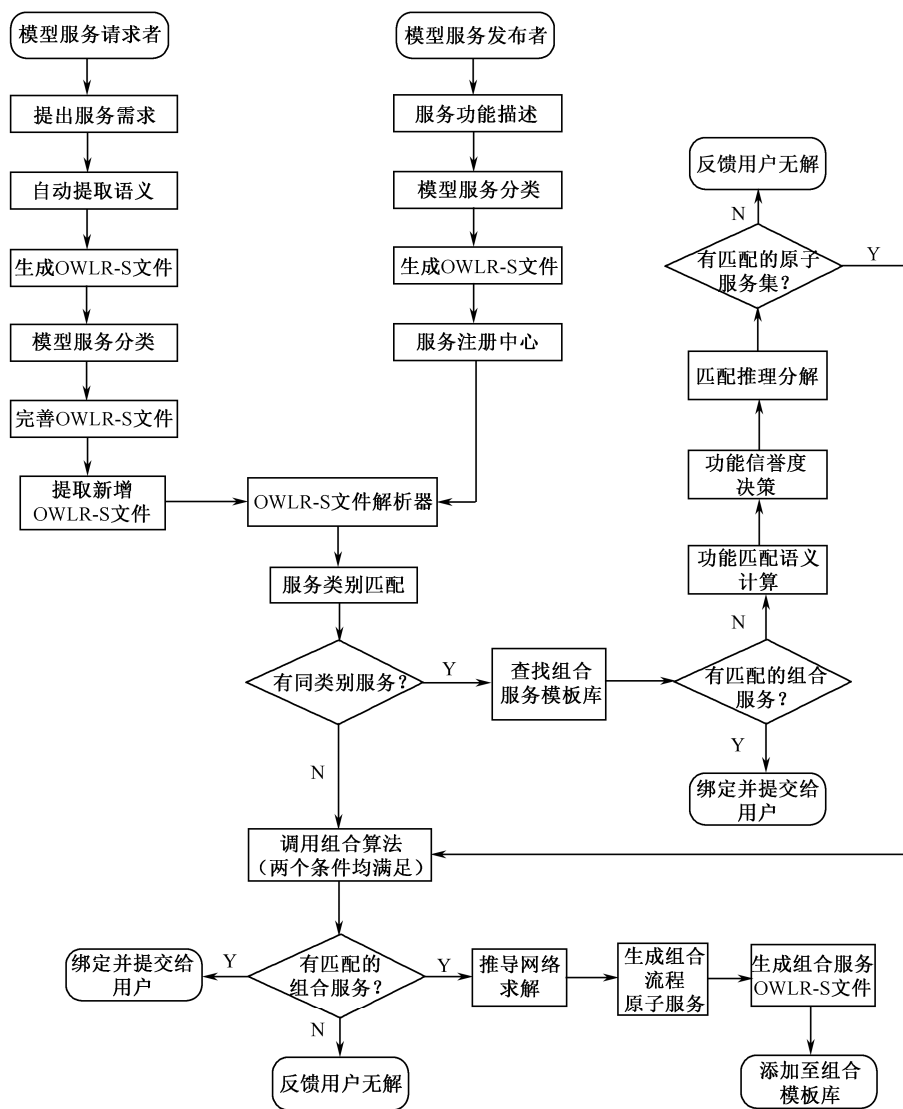


图 5-2 组合服务执行的流程图

Step4. 模型服务发布者把生成该服务的 OWLR-S 文件在服务注册中心进行注册，供模型服务请求者查找。

2. 模型服务请求者

Step1. 服务请求者根据仿真实际，结合作战仿真领域本体，提出仿真服务

需求,系统根据请求者的需求自动提取语义,并对服务的功能进行语义描述生成 OWLR-S 文件。

Step2. 生成的 OWLR-S 文件提交给服务分类模块。服务分类模块根据申请服务的信息,对申请服务进行分类。

Step3. 将 Step2 得出的分类信息写入 OWLR-S 文件,作为服务需求信息扩充到 OWLR-S 文件,为模型发布者提供需求信息。

3. 推理与组合过程

Step1. OWLR-S 解析器对服务请求者和服务注册中心中服务发布者注册服务的 OWLR-S 进行解析,提取出服务的类别信息、功能信息和信誉度信息。

Step2. OWLR-S 解析器得出请求服务的类别信息,并找出服务中心中与请求服务属于同一类的发布服务。

Step3. 若服务注册中心存在与请求服务同类别的发布服务,则把这些发布服务作为候选匹配服务交给组合服务模板库,转至 Step4;若服务注册中心不存在与请求服务同类别的发布服务,则转 Step10。

Step4. 根据请求服务的需求,对组合模板库的组合服务进行功能匹配,其中概念相似度的计算是功能匹配的基础,而概念相似度又是基于领域本体来计算的。

Step5. 若组合模板库中存在满足请求服务需求的组合服务,则将有效的组合服务返回给用户;若组合模板库中不存在满足请求服务需求的组合服务,则执行 Step6。

Step6. 对由 Step3 得出的与请求服务同类别的发布服务进行功能匹配,得到服务的功能相似度值和服务接口之间的语义关系。

Step7. 对由 Step3 得出的与请求服务同类别的发布服务进行信誉度决策,得到服务的信誉度决策值。

Step8. 根据 Step6 和 Step7 得出的功能相似度值、服务接口之间的语义关系和信誉度决策值,并进行匹配推理。

Step9. 若不存在满足用户需求的原子服务集,则告诉用户推理过程结束;若存在匹配的原子服务,则转至 Step10。

Step10. 若 Step9 中存在满足用户需求的服务集,若服务注册中心不存在与请求服务同类别的组合服务,运用算法进行服务组合分析。

Step11. 经过 Step10 分析后,若存在满足请求服务需求的组合服务,则将有

效的组合服务和服务规则集合 RS 返回给用户并转至 Step12; 若不存在满足请求服务需求的组合服务, 则告知用户无满足要求的组合服务。

Step12. 根据服务规则集合 RS 使用推导网络求解算法, 得到组合服务的图形流程。

Step13. 把组合服务的组合流程转化为原子服务的形式。

Step14. 根据组合服务中各个原子服务的 OWLR-S 文件, 得到组合服务的 OWLR-S 文件。

Step15. 把组合服务的 OWLR-S 文件交由服务组合模板库, 对模板库进行更新, 结束。

5.4 服务组合示例

5.4.1 仿真服务基本需求

实例背景: 命令某导弹旅防空营一辆导弹发射单元向可疑目标进行机动并启动传感器, 传感器获取 Track 信息后, 首先会判断探测到的战场实体是否为敌军。如果是, 则将敌军的位置等信息报告给上级指挥所并停止机动任务, 等待上级进行决策 (等待支援或撤退); 否则, 不做任何处理继续执行机动任务。指挥所收到报告信息后, 利用雷达探测设备探测战场情况, 主要是查看在发送方 15km 范围内是否存在其他友军力量。如果有, 则命令其中一个导弹发射单元机动至发送方位置。当被指派的支援单元到达发送方位置后, 启动火力控制单元进行打击目标分配, 并向各导弹发射单元下达准备攻击命令, 导弹发射单元实时向指挥所上报敌方目标的具体情况。当目标在导弹发射单元的探测范围并具备攻击条件时, 指挥所向火力控制单元下达攻击命令, 两辆导弹发射单元分别对目标进行火力打击。

5.4.2 模型组件设计分析

该问题可抽象的实体模型为指挥所、导弹发射单元、目标、火力控制单元和

雷达探测模型。指挥所应具有通信、决策（火力突击或撤退）、指派下级执行作战或撤退等任务的能力；导弹发射单元应具有探测战场实体、上报、通信、等待支援或撤退等能力；目标可作为一个最基本的实体存在同时具有可被探测的能力。火力控制单元应具有目标分配和控制组件，具有能向导弹发射单元下达攻击命令的能力，输入参数目标类型、数量、威胁指数、相对速度、位置、航向等，输出信息包括打击时间、武器、数量等。雷达探测模型主要用于探测并识别敌方实体，其输入参数主要有相对距离、方位角、速度、天气影响因子、干扰因子及雷达反射截面积等，输出参数为位置、速度、航路角、数量、类型、探测时间、最早威胁时间、威胁指数等。

上述分析涉及 5 个实体模型，这 5 个模型实际上对应于 5 个模型组合服务。每个组合服务由不同的原子服务构成，每个原子服务分别对应原子模型组件，因此，涉及的原子模型组件包括指挥所平台组件、地面机动组件、通信组件、导弹发射单元平台组件、传感器组件、目标平台组件、目标分配组件和控制组件等。

假设服务注册中心中存在各种类型的原子组件封装而成的原子服务，同时存在目标模型服务和指挥所实体模型服务，但不存在火力控制单元模型服务、导弹发射单元实体模型服务和雷达探测模型服务，因此目标模型服务和指挥所实体模型服务可以直接进行绑定和调用，而其他 3 种模型服务需要由多个原子组件组装而成并以组合服务的形式提供。

按照组件化建模的基本原则和军事概念模型的基本构成，将各实体模型划分成由组件装配而成的各种组合形式：

指挥所实体模型=指挥所平台组件+地面机动组件+通信组件；

导弹发射单元实体模型=导弹发射单元平台组件+地面机动组件+通信组件+传感器组件；

目标模型=目标平台组件+地面机动平台组件；

探测雷达模型=通信组件+传感器组件；

火力控制单元=目标分配组件+控制组件。

5.4.3 模型组合建模

组件对应原子服务，各实体（目标、雷达探测、火力控制单元）对应基本服务。如果用户需要在一次服务中体现上述的所有过程，则需要通过一种组合服务完成，即需要对组合模型进行分析。首先需要完成对基本服务的分析，考虑到基

本服务是作战仿真中比较常用的，假定类似这样的服务在注册中心直接存在，这里重点对复杂的组合服务进行分析。考虑到目标模型和指挥所实体与探测雷达、火力控制单元、导弹发射单元均有联系，目标模型和指挥所实体不再展开，但与探测雷达、火力控制单元、导弹发射单元具有组合关系，各模型之间的接口关系如图 5-3 所示，这种关系实际上是一种逻辑关系，更多的是体现接口关系、数据传递关系。

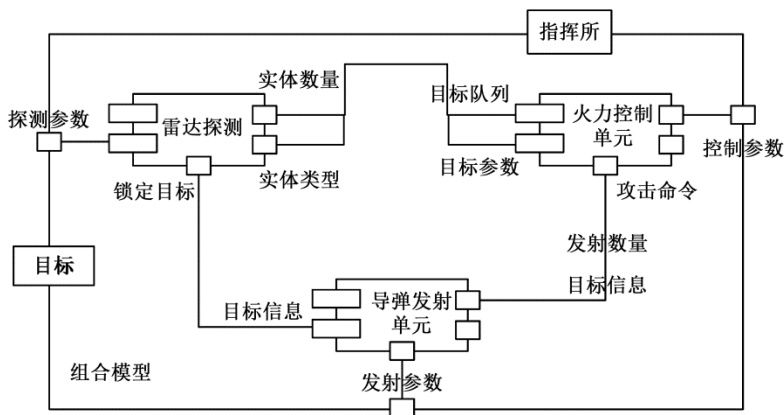


图 5-3 组合模型建模

5.4.4 链式服务组合建模

从模型组合的角度分析，首先根据用户提出的仿真需求服务进行组合性解析，将需求按照语义关系进行划分和抽象，找出满足组合要求的所有子模型（雷达探测模型、目标模型、指挥所实体模型、火力控制单元模型和导弹发射单元模型）。根据模型接口和功能的语义描述，进行语义提取与转换，同时与模型知识库进行比对，并将提取的语义转换成组合推理引擎能够识别的语言，在组合规则库和模型知识库的共同作用下，按照作战行动的过程，遵循接口匹配规则、组合规则和约束规则，对模型组合流程进行建模，推理引擎按照模型组合服务流程对模型进行组合建模^[58]，形成一个组合的模型，确立各模型之间的接口关系。然后将各个基本服务的执行顺序、数据传递关系和接口连接关系等以服务的方式进行描述和封装，并进行模型组合性验证。最后将组合结果以服务的方式提供给用

户直接使用。

从服务组合的角度分析,服务组合的任务是发现一组服务可以满足服务请求,组合结果可能是一个服务或一个服务链,也可能是一个带有分支和汇合结构的服务集合。假设 DS_1 、 DS_2 、 DS_3 和 DS_4 分别表示导弹发射单元平台服务、地面机动服务、通信服务和传感器服务,按照服务组合的需求封装组合成导弹发射单元模型服务; ZS_1 、 ZS_2 和 ZS_3 分别表示指挥所平台服务、地面机动服务和通信服务,根据指挥所运行需求封装组合成指挥所实体模型服务; MS_1 和 MS_2 分别表示目标平台服务和地面机动平台服务,封装组合成目标模型服务; LS_1 和 LS_2 分别表示雷达通信服务和传感器服务,封装组合成雷达探测模型服务; HS_1 和 HS_2 分别表示火力目标分配服务和控制服务,封装组合成火力单元控制模型服务。

如图 5-4 所示,是模型组链式服务组合示意图。服务组合的基础是首先查找出满足用户需求的一系列服务,在仿真实例中主要包括导弹发射模型服务、雷达探测模型服务、目标模型服务、指挥所实体模型服务和火力控制单元模型服务。而后将这些服务以过程的形式构造成为更为复杂的服务,并采用 BPEL4WS 语言对服务链中的每一个服务执行顺序、服务之间的控制关系、数据流进行详细描述,利用 `partner` 指定服务链中的每一个服务,同时使其指向服务对应的 WSDL 文档^[59],通过 `sequence` 确定服务链执行逻辑(顺序模式)。同时,也要建立该过程的 WSDL 文档,以便其他应用或服务调用。图 5-4 中只是一个逻辑结构,实际上在服务流程构造时,每一个子服务可能会被多次调用,也可能是以并发的方式或者是循环的方式在整个服务链中被执行,直至服务链的业务流程构造完毕。服务链的业务流程构造是一个复杂的过程,是由计算机根据服务语义关系、描述逻辑、分析推理后自动形成的。最后,将 BPEL4WS 的描述过程以 WSDL 文档形式封装成组合服务供用户调用。

在形成服务组合链的基础上,对于服务组合建模过程而言,每一个基本服务模型之间存在接口关系、执行顺序和数据传递等服务业务流程建模^[60],服务组合的逻辑建模如图 5-5 所示。

在组合服务的内部,逻辑上划分为 8 个服务交互调用的子过程。这 8 个基本过程分别是:①导弹发射单元模型服务与目标模型服务的交互;②导弹发射单元模型服务与指挥所实体模型服务的交互;③指挥所实体模型服务与雷达探测模型服务的交互;④雷达探测模型服务与目标模型服务的交互;⑤导弹发射单元模型服务与雷达探测模型服务的交互;⑥指挥所实体模型服务与火力控制单元模型服

务的交互；⑦火力控制单元模型服务与导弹发射单元模型服务的交互；⑧导弹发射单元模型服务与指挥所实体模型服务的交互。因此，对于一个较为复杂的组合服务而言，它可能是多个基本服务的组合，而每个基本服务之间在整个过程又需要交互，基本服务之间的交互是通过服务接口访问实现的，主要是提供服务的输入、输出参数。前一个基本服务的输出参数作为下一个基本服务的输入参数，如此循环往复下去，直至服务组合链上最后一个服务结果的输出。

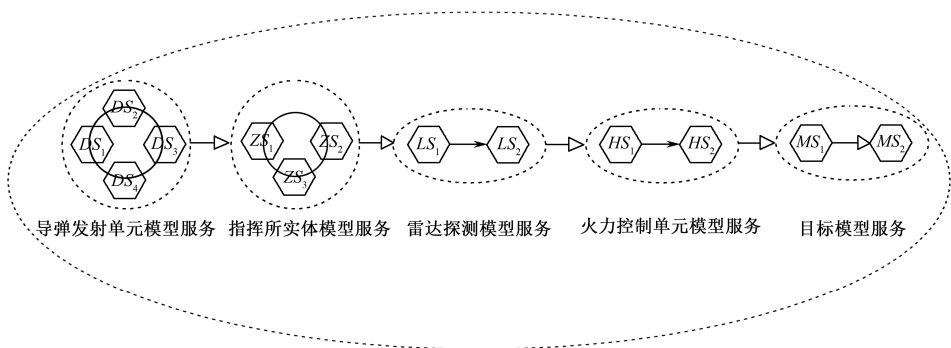


图 5-4 模型组链式服务组合示意图

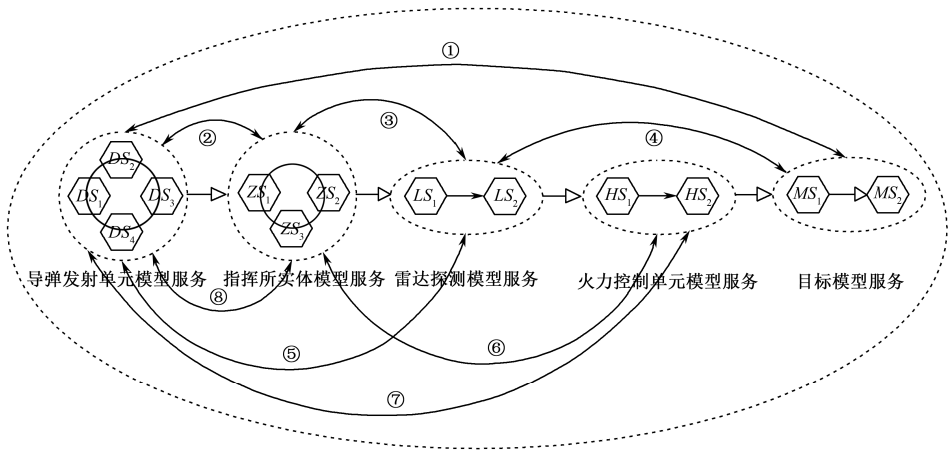


图 5-5 服务组合逻辑建模

5.5 本章小结

本章主要从模型组合的基本概念出发,将模型服务分为原子服务、基本服务、复杂服务和组合服务四种类型,通过建立模型服务匹配模型实现服务的匹配功能,分别从服务的发布者、服务的请求者和推理过程这三个角度对组合服务的执行流程进行具体描述。在此基础上,通过构建服务组合语法、组合规则、组合算法提出了一种面向服务的作战仿真模型组合建模方法,采用过程构造子按照模块化方法对模型服务的组合方案进行描述,解决了服务组合域表示困难、描述能力不足、模型匹配差异性大等问题。最后,通过服务组合实例实现了一种链式服务组合建模方法,说明了模型组合的具体过程。

第 6 章

面向服务的模型集成框架 设计与实现



服务是接口的实体化表示,通过接口的约定和协议向外部提供各种操作和功能。作为一种设计方法,面向服务的方法与面向对象存在较多的差异。面向对象的方法是从对象的角度进行设计,主要任务是通过构建静态和动态关系实现对对象之间的操作和访问。面向服务则通过接口的设计实现服务的访问,将系统中的任何对象及其产生的行为都建模为服务,对象之间的访问和操作通过服务的方式完成。面向服务的方法是一种具有很好扩展性的方法,向用户屏蔽了具体实现的细节^[61]。采用面向服务的技术架构 SOA (Service-Oriented Architecture),以仿真服务来组织和管理仿真模型资源,所有模型在 UDDI 注册为服务,通过服务发现、服务匹配等环节将仿真模型服务按需组合后提供给分布式仿真节点,实现模型的快速重构,以满足特定用户需求的仿真。这种灵活的重构方法对于分布式环境下建模和仿真的重用和组合具有重要作用。

采用服务的方式实现模型的集成应用的目的在于提高仿真模型的重用性和组合性,将功能独立的模型组件封装成标准的模型服务,并通过平台无关的通用协议进行服务调用,按照服务规范对仿真模型服务进行描述并发布,通过服务发现和服务选择来动态构建满足实际需求的更为复杂的组合服务,以满足不同需求的仿真应用,实现模型的快速重构和柔性组合,提高模型重用率和开发效率。

6.1 需求分析

以陆军作战仿真模型为基础,构建仿真模型集成框架,实现对模型的组合性应用,是实现陆军作战实验、武器装备需求论证研究的有效手段,为全面推进陆、海、空及联合作战实验提供可重用、可组合的模型服务积累经验、探索路子。

经过多年来作战仿真与建模的积累,在各军兵种院校、科研院所已经相继研制出各种类型的作战仿真模型。从体系构建的角度来看,模型的构建研究可以归纳为“四个层次、六个军兵种类的六类模型”,即联合战役、军种战役、合同战术、分队战术四个作战规模层次,联合、陆军、海军、空军、火箭军、战略支援六个军兵种(专业)作战行动,武器装备、作战行动、指挥控制、战场环境、综合作用和分析评估六种模型类型。以这种立体交叉的仿真模型体系为基础,同时与陆军部队武器装备训练、模拟训练模式同步,各单位不断完善现有模型及开发新的仿真模型,形成了能够适应我军作战实验需求的仿真模型体系。针对各兵种、各类仿真模型资源,探索能够综合利用和分类管理的开放式平台,对于综合集成、整合各类型仿真模型,使其成体系化、规范化,并具有可集成、可互联、可组合的应用特点,便于众多作战仿真模型在统一的软件框架控制下,实现动态重构和灵活组合,以满足不同的仿真应用。

目前,虽然国内外已经进行了大量卓有成效的仿真模型集成研究工作,而且也初步建立了一些具有代表性的仿真模型集成框架,但也应该看到:大部分的建模与仿真活动都没有系统考虑资源的共享,不同的仿真系统往往相互独立,严重影响到建模与仿真的互操作性、可重用性和可信性,增加了不必要的开发和存储冗余。对于仿真模型的集成研究,主要集中于采用 MDA (Model Driven Architecture) 技术、组件化技术、Web 服务技术、SOA (Service-Oriented Architecture) 技术和可扩展框架技术等。当前在 M&S 领域,基于 MDA 的思想已经用于许多中小规模的项目和公用集成框架中。在 XML (eXtensible Markup Language) 标准的支持下,可以有效描述仿真设计模型信息,支持仿真实验和想定配置,并通过 Web 服务有效检索、查询和管理仿真模型组件信息,从而更好地支持仿真模型的工程化开发与集成^[62]。

美国海军研究生学院等多家研究机构及公司联合开展了可扩展建模仿真体

系框架 XMSF (eXtensible Modeling & Simulation Framework) 项目的研究。XMSF 通过尽量采用大量成熟的商业化标准和框架, 如 Web Service、XML、X3D (Extensible 3D)、UML 等进行仿真应用系统之间的松散集成, 通过包含相关元数据和元模型的资源仓库对模型资源进行管理。

事实上, 现有的模型集成框架并不能为大规模的仿真模型重用和组合提供全面支持, 主要存在以下几个方面的问题。

1. 理论方法支撑不足

模型的重用和组合强调自动选择和动态组合, 在分布式环境下, 在语义和语法层, 模型集成框架难以实现对模型的准确描述和自动理解, 对于模型服务的形式化描述缺乏全面的领域本体和描述逻辑, 目前还没有一个完善的、针对军事仿真领域的本体库。这就在对模型的描述过程中, 使得计算机在识别上有偏差, 基于本体的描述逻辑的方法还不能很好地解决军事仿真领域的诸多问题, 难以为更高和更多层次的模型组合提供支持。特别是仿真模型的重用和可组合性缺乏相应的方法论指导, 更缺乏可实施的工程化方法。

2. 模型组件服务描述不足

服务发现是模型重用和快速重构的关键, 而服务描述是服务发现的基础, 商业化的服务描述方法经过多年的积累和应用已经相对完善, 而由于军事领域的模型服务描述起步较晚, 且应用规模也只是在个别单位和小范围内, 缺乏标准的描述规范。同时, 现有的可组合仿真工程方法并不注重仿真组件的选择, 主要是通过元数据实现相关检索, 缺少智能化分析, 使得在组合性分析中难以自动推理。即使是规范的概念模型描述方法也不能自动转换成机器可理解、可识别的信息, 服务的接口语法和静态功能语义的描述能力限制了对仿真模型动态行为的表示, 往往在模型服务选择过程中出现偏差。

3. 模型组合语义方法不足

由于作战仿真模型相对复杂, 同一种模型可能因为开发单位的不同而存在较大差异, 一是没有统一描述规范; 二是由于封装后模型之间难以实现理解上的一致性, 使得在语义理解出现偏差, 而影响组合效果。同时, 由于军事仿真服务与其他服务相比, 特别是在服务组合过程中, 更加复杂, 存在多种关系, 如指控关

系、隶属关系、继承关系、通信关系等,对这些关系的描述缺乏有效的方法和手段,对于关系的描述和推理难以满足实际要求^[63]。

随着仿真技术的不断发展和仿真规模的日益扩大,实现模型资源的可重用变得越来越迫切,仿真模型集成和组合应用需求也越来越明确,其迫切的军事需求主要体现在以下几个方面:

(1)可以有效指导模型开发,减少重复建设,提高模型共享服务水平,以满足不同决策层次、决策对象对作战仿真推演过程进行有效控制的需要,为诸军兵种在时间、空间、信息、任务等方面的作战协同提供模型支撑。

(2)通过描述逻辑、本体映射、服务描述等方法实现模型封装,有助于作战模型资源的集成、组合和管理,实现作战仿真模型体系构建过程中主要环节的规范化、标准化,满足服务访问的统一性。

(3)通过服务组合的方式实现不同粒度的模型组合和集成,实现模型的多层次动态集成与重构,解决不同分辨率、不同层次的模型组合和集成调用问题,实现作战仿真实验所需的多分辨率异构模型互联、聚合、解聚与联合运行。

(4)实现模型开发、存储与共享的一体化,将分散的、不同粒度的模型统一、科学和规范管理,最大限度地保证模型的重用,充分发挥模型的军事效益,避免重复投资、重复建设,提高模型的可信度和互操作性。

针对上述存在的问题和面临的迫切需求,基于 SOA 的仿真体系结构和建模仿真框架,通过对模型服务描述、服务发现、服务组合和集成应用等进行研究,以更好地满足大规模仿真模型资源共享的需求。

6.2 基于 SOA 的集成框架分析

框架对于相似的问题能够提供一种统一解决方案,仿真模型集成框架首先应该是一个开放体系的体系结构,其目标是能够动态地组装构件,并将构件以服务的方式注册,然后向各分布式用户提供仿真模型服务。采用基于 SOA 的架构实现框架集成,基于 Web 服务标准,从更一般的服务概念出发,通过重构和组合模型服务来满足不同用户的仿真需求。

在作战仿真模型组件库的基础上,构建基于 SOA 的陆军作战仿真模型柔性集成系统,实现模型构件库、建模知识库和模型数据库的统一组织与管理,为模

型资源集成与重用奠定基础。从目前的需求来看,这样的模型资源管理系统必须采取一种分布式的管理方式,因此需要解决好下面两个问题:一是模型库动态管理问题;二是柔性集成框架设计问题。

面向服务体系结构 SOA 能够解决分布式环境下模型集成的需要,是一种通过网络互联能完成特定仿真任务的独立功能实体实现的软件体系结构。通过服务之间定义良好的接口和协议联系起来,使构建在各种系统中的服务能以一种统一、通用的方式进行交互。SOA 将功能封装成服务,对外只提供了定义的接口,而屏蔽了内部功能实现的细节,能提高模型功能的重用性和组合的快捷性。

SOA 是一种松散耦合的软件架构,它可以根据需求,通过网络对松散耦合的仿真模型组件进行分布式部署、组合和使用。该架构主要有以下特点:①松散耦合。在 SOA 中,服务提供者和使用者是相互独立的,通过 UDDI 建立相互间的联系,服务请求和服务访问之间没有严格的约束关系,是一种松散耦合关系。②透明服务。服务需求者只需要描述所需服务的功能,服务注册中心会根据相关需求自动分析并提供相应的服务,服务使用者也无须关心服务实现的具体细节和服务存放的具体位置。③自治组合服务。服务是由模型组件封装而成的组合模块,并以组合功能实体提供独立服务。④标准化服务接口。服务之间通过既定的通信协议进行互操作,按照标准化 Web 服务描述语言 WSDL (Web Services Definition Language) 接口提供应用功能,并可基于标准化传输方式,采用标准化简单对象访问协议 SOAP 进行调用。

通过对以上特性分析可知,SOA 架构能为模型集成框架提供更加灵活的构建方式,基于 SOA 构建集成框架,可以从底层架构的级别来保证仿真应用的松耦合性及灵活性,适应不断变化的仿真应用环境,实现模型的按需服务。

基于 SOA 的思想可以为不同组织、不同领域提供仿真模型服务、仿真应用服务、指挥控制服务等,从而通过重构和组合模型服务方式满足不同用户的仿真分析、仿真实验、仿真训练和仿真论证需求。结合陆军作战仿真的需要,通过对 SOA 的分析,构建的基于 SOA 的仿真模型集成框架具有以下优势。

(1) SOA 具备较为成熟的开放标准,在服务组合和服务集成应用方面具有很好的应用实例,能够满足当前仿真用户的实际需求。

(2) SOA 支持分布式部署和访问方式,具备大规模网络环境下的模型快速重构和组合的能力,减少仿真模型的重复开发,对于分布式仿真而言能够提供快速建模能力,模型间的互操作性和可扩展性增强,提高仿真应用的层次和水平。

(3) SOA 能够提供模型的自治管理和动态更新机制,模型的共享水平得到

提升,使得建模仿真的专业化、标准化程度增强,进而能解决模型、仿真、环境和应用表示的一致性问题,提高模型组合的有效性。

(4) 基于标准化的服务接口,能简化模型交互过程中的复杂操作,提高模型访问效率,能够对已有模型和新开发模型进行有效组合和集成,提高建模仿真的可伸缩性和灵活性。

基于 SOA 的模型集成框架的目标是快速构建和组合模型,为建模仿真人员提供模型服务开发、注册、部署、发布、发现和组合,使模型运用具有一定的规范性、灵活性和可扩展性。基于 SOA 的集成框架需具备如下能力。

1. 快速重构能力

重构是组合的基础,模型、接口和应用开发等需要采用构件化方式、标准化接口,使得重构后的模型具有良好的互操作性,实现上下贯通,支持多层次、多级别的模型应用。模型的集成应用通过服务的方式实现,模型服务的描述、调用和发现分别采用 WSDL (Web Services 的描述技术)、SOAP (Web Services 的调用和交换技术) 和 UDDI (Web Services 的发现技术) 实现,这些是进行快速重构和组合的技术基础。重构的目标是实现模型服务能力、服务交互关系和服务应用配置的重新构建,以便于在现有模型基础上实现更为复杂的功能,避免重复建模。

2. 灵活组合能力

在面向服务的集成框架下,任何模型的应用本质上均是通过服务组合得到的,服务组合实际是对原子服务、基本服务和复杂服务之间的相互组合^[64]。对灵活组合建模的要求如下:①应该建立一种有效的组合建模方法,只需开发一些原子模型、基本模型,复合模型可以通过组合方式获取;②能够以即插即用方式组合模型,且能够易于验证组合模型的有效性;③能够根据业务模型的逻辑推理自动创建过程模型,并形成可以执行的业务工作流程,直接供服务模型执行。

3. 动态调度能力

模型服务动态调度包含虚拟空间分配与服务执行两个阶段,虚拟空间分配主要针对不同的仿真任务提供多种调度策略,负责模型服务的统一调度,实现与各管理站点模型资源调度器的协作,完成模型服务定位与分配、进程创建和任务迁移等。良好的调度策略对于改善和提高模型仿真效率具有重要作用。服务执行主

要是通过标准的服务协议进行交互,并根据仿真用户需求使用相应的分布式仿真调度策略执行仿真服务。特别是当组合服务集不在同一站点,服务须由不同的站点向同一个用户终端提供时,动态调度就需要根据执行顺序进行服务调度。

4. 服务管理能力

在 SOA 体系架构下,服务描述与服务实现是相互分离的,这就给服务管理带来了一定的困难,因此,模型服务的管理主要是针对描述与实现的管理。服务描述必须能够满足服务组合算法的查询要求,并建立服务描述解析模型,主要对服务信息、功能信息和状态信息等进行自动分析推理。服务实现管理主要是对服务的注册和部署、分类管理、服务发现及组合方案的选择等,需要为仿真模型的组合建模过程提供自动化支持。

5. 数据管理能力

仿真开发和运行过程中会涉及多种应用相关的文档和数据,如组合服务描述文档、仿真应用想定配置、仿真运行和结果数据。这些不同形式的文档需要通过仿真应用管理工具进行统一组织管理,并能以有效的方式向模型服务提供所需的数据,这是进行模型正确输入/输出的前提。

基于 SOA 体系架构,采用“资源+服务+平台+应用”的柔性设计思想构建作战仿真模型柔性集成框架。柔性体系结构设计基于一套实用的系统构建技术标准和工具,按照这些标准研制的各种构件相互可以即插即用,这种方式可以使仿真模型根据联合作战任务要求进行灵活重组,并形成最佳的组织结构。在该集成框架下,将模型的重构、装配、封装、组合和调用等以服务的方式呈现,根据用户定制需求,将各种服务动态组合起来,快速为用户提供个性化的仿真模型和组合服务,实现模型的重构、组合和集成调用。

面向服务的体系结构 SOA 将应用程序的不同功能单元分别作为服务,采用松散耦合的方式对服务和应用进行集成。服务之间的交互通过接口和协议完成,按照分布式构造方式,将模型功能以服务的形式提供给最终用户或供其他服务调用,实现服务组合、集成与管理,隐藏了服务内部细节和交互数据的复杂性,即使是在不同的计算机上,采用不同的操作系统和建模工具开发的组件,都统一采用服务进行封装和描述,降低组合和集成的复杂性,增加作战仿真应用的灵活性。

基于 Web 服务的 SOA 的关键是使用标准的服务接口和松耦合的连接,分为服务发布、服务管理和服务调用等,具体如图 6-1 所示。其实现过程如下:①建立服务注册中心,实现服务的发布和管理。②服务发布者提供模型服务描述文档(包括 SSProfile、SSModel 和 SSWSL),并在 UDDI 中心完成注册,对入库模型业务逻辑进行封装,实现统一接口。③服务注册成功后,在 UDDI 和模型库之间建立映射关系。UDDI 中心并不存放实际的模型,而是关于模型描述的信息以及访问模型的位置信息。④服务使用者提交服务请求。通过语义抽取和组合分析后,经过解析被封装成 SOAP 消息,读取 UDDI 中的模型服务信息。⑤按照服务发现、服务匹配和服务组合的过程获取服务句柄。⑥建立服务绑定关系,由服务代理调用服务,直接提供给用户。

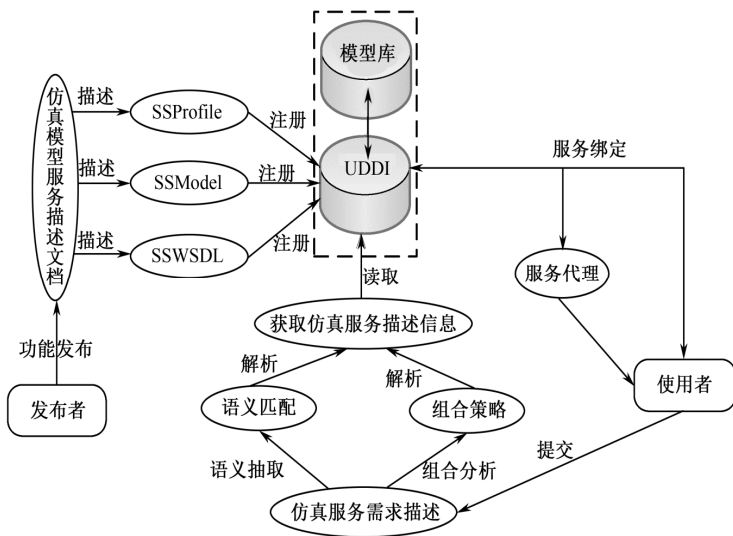


图 6-1 服务发布、服务管理与服务调用过程

图 6-2 描述了基于 SOA 的仿真服务开发流程。在 SOA 框架下,模型服务由服务实体和服务描述组成。服务实体具有一定功能的服务,通过注册中心部署在服务器中。服务描述是对服务所能提供功能的具体描述,是进行服务发现和服务组合的基础。

下面分别从用户和开发者两个角度对模型服务的设计开发过程进行详细描述。首先,建模人员根据建模仿真需求明确要解决的问题,对问题抽象构建仿真域中各要素的关系,并建立军事概念模型,技术开发人员依据军事概念模型设计

并实现组件模型，通过模型组合和装配后进行相应的 VV&A 测试。在此基础上，根据需要将仿真模型封装为原子服务、基本服务、复杂服务和组合服务，将模型服务部署到相应的服务器，生成相应的服务描述文档并在 UDDI 中心进行注册，同时向服务代理提交后进行服务发布。

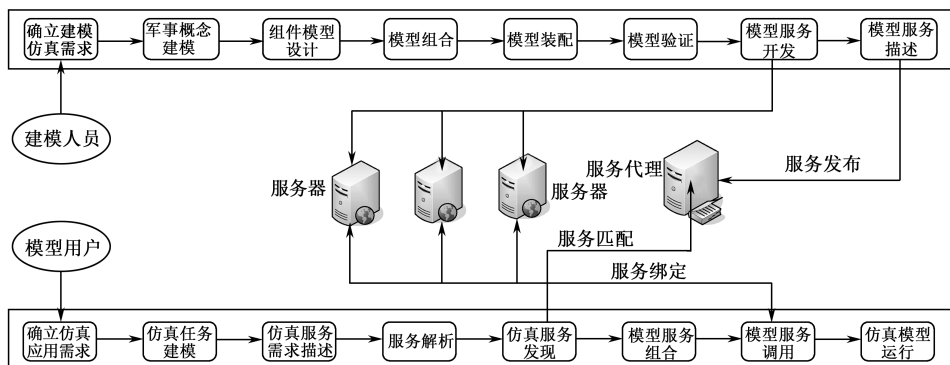


图 6-2 基于 SOA 的仿真服务开发流程

其次，仿真用户根据仿真应用需求，进行仿真任务建模，并在此基础上转换为仿真服务需求描述，通过服务解析、组合策略分析等手段在 UDDI 中发现满足需求的仿真服务子集，经过服务匹配、服务选择和服务组合等环节确定仿真服务，完成相关服务参数配置后提交至服务代理，通过服务绑定实现模型调用与运行。

由上面的分析可知，在面向服务的体系架构支撑下，仿真应用的分解不是按照以往的实体分解方法，而是按照“功能”的分解方法，首先选择满足功能需求的仿真服务集合，然后根据语义匹配和逻辑推理分析确定模型组合方案，完成特定的仿真应用服务。这种集成框架能够提供更细重用粒度的仿真模型，将模型功能转化成了服务能力，为开展更好服务的仿真应用提供了基础条件。

6.3 基于 SOA 的仿真模型集成框架总体设计

采用基于 SOA 架构的方法来实现作战仿真模型集成框架，在该集成框架下，基于 MDA 方法实现模型的统一转换，采用组件化技术实现对模型的重构、装配，

并以服务的方式实现模型封装、组合和调用等，使不同军兵种模型、不同层次模型和不同粒度的模型以一种通用、统一的方式进行交互和共享。参照面向服务的体系结构，采用分层描述的方法，设计一种基于 SOA 的仿真模型集成框架 IFSM-SOA（Integrated Framework of Simulation Model based on Service-Oriented Architecture），将仿真模型集成框架分为四层，分别是资源层、构件层、服务层和应用层，其具体结构如图 6-3 所示。

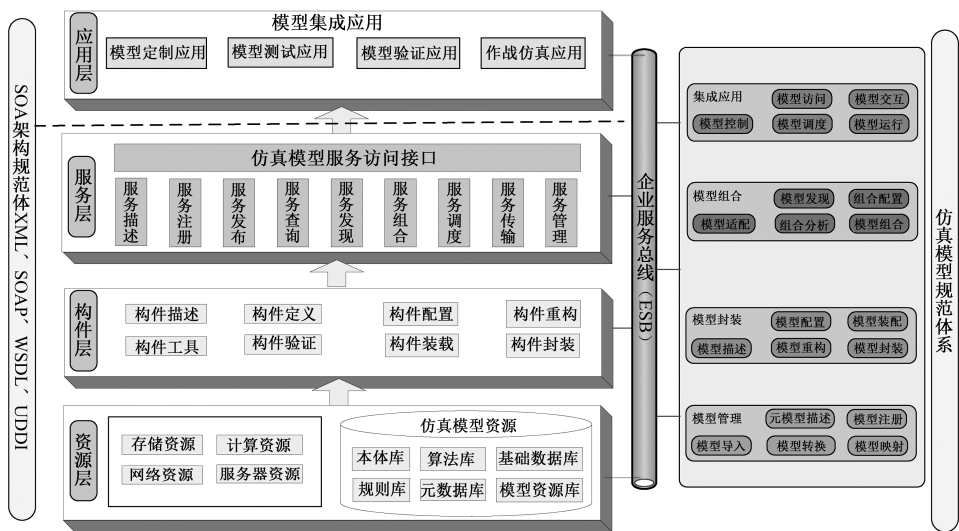


图 6-3 面向服务的作战仿真模型集成框架总体结构

6.3.1 资源层

资源层主要解决资源的描述、组织和管理等功能，包括计算资源、设备资源、服务器资源和本体库、知识库、模型库等数据资源，该层是集成框架的基础。模型库主要用于存储各类模型，是系统的核心部分，在逻辑上模型库是各种模型的集合，在软件内容上，则由许多计算机程序组成^[65]。模型库采用关系数据库的组织结构形式，在数据库的表空间中存储模型和相关文档的标识信息、模型名称、类型、性质、所属军兵种、功能描述等信息。模型导入需要按照一定的规范完成，主要包括以下内容。

- (1) 模型标识信息：模型标识码、名称、类型、军兵种类别等；
- (2) 模型详细信息：模型功能描述、应用层次、创建时间等；
- (3) 模型类别信息：模型类别标识码、类别名称等；
- (4) 模型调用接口信息：接口参数类型、参数名称、访问方式等；
- (5) 模型组合关系信息：父模型标识码、子模型标识码、调用次序等；
- (6) 模型文件信息：模型文件标识、文件存储路径、创建时间等。

6.3.2 构件层

将不同层次、不同类别、不同粒度的模型以组件的形式进行封装，使模型库在组织形式上表现为可重用的模型组件库，因此，这里所说的构件实际上可以理解为由组件。可重用的模型组件库不同于以前的模型对象库，模型组件库保存的是一些经过测试的模型组件。这些模型组件遵从一定的标准或规范完成对各种模型的统一描述，并生成模型组件描述文档。组件层主要包括组件描述、组件定义、组件构建工具、组件验证、组件配置、组件装置、组件重构和组件封装等，其主要任务是完成模型的描述、重构和装配。

构件层的核心是通过基于接口的程序设计完成对入库模型的标准化描述。每个组件都会提供一个定义好的接口，可以与其他组件进行交互。用户或其他应用程序可以使用这些接口将模型库中的模型应用到自己的程序中，与其他组件或应用程序联合运行。

6.3.3 服务层

采用 Web 服务的方式实现对模型的封装，能够屏蔽不同层次、不同粒度、不同军兵种模型间的异构性，便于统一运用和共享。Web 服务定义了如何描述、访问服务及如何发现和验证相关的服务。Web 服务层由服务描述、服务注册、服务发布、服务查询、服务发现、服务组合、服务调度、服务传输和服务管理等组成，在屏蔽下层资源异构的同时便于为应用层提供更高级的服务，其中 Web 服务描述及操作交互如图 6-4 所示。

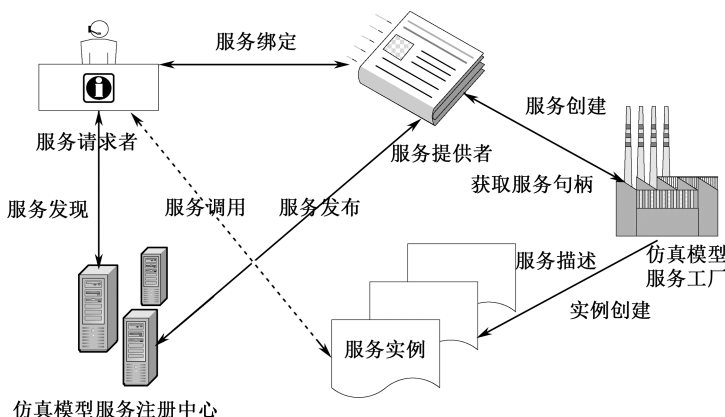


图 6-4 Web 服务描述及操作交互

服务发布：服务提供者定义 Web 服务的服务描述，Web 服务通常在 UDDI（Universal Description Discovery and Integration）中心进行注册发布。

服务发现：服务发现指通过一定的方式找到满足特定需要的 Web 服务过程。UDDI 提供了描述服务的框架，可存储大量 Web 服务的描述性信息，用户通过 UDDI 中心可以查找所需要的服务及服务调用的方法。

服务描述：Web 服务具有自描述性。定位了一个服务后，可以通过它自身的描述来获取服务所提供的操作及调用这些操作的具体方法。服务使用 WSDL 语言描述，能提供一个基于 XML 的简单语汇表，用来描述 Web 服务能做什么、在何处及如何调用。

服务调用：Web 服务的调用涉及消息在用户端和服务端的传递，客户端使用 WSDL 文档中的信息，通过相应机制调用该 Web 服务实例所提供的操作。

通过服务的组合实现模型的组合，主要包括模型发现、模型适配、模型选择、模型组合和模型组装等过程。

（1）模型适配：①经过构件组装后，模型被封装成原子服务，并对每个原子服务模型进行规范化描述，在此基础上提取其语义；②分析待匹配的模型（原子服务）的语义信息和数据类型信息，分别得出语义信息相似矩阵和数据类型相似矩阵；③运用服务匹配算法进行语义相似性运算，得出匹配相似矩阵，生成新的模型。

（2）模型选择分为：模型自动选择和人工选择两种。

其中，模型自动选择包括：①依据标准化描述规范，提取所需要的复杂模型

(组合后的模型)特征信息;②通过人工智能识别器,以模型类型选择规则库的规则为依据推理,对模型构件库中的模型进行识别、选择;③提取模型参数并从数据库中获得相应的数据。

模型人工选择包括:①根据实际需要,查询到满足条件的模型;②在列出的模型列表中进行勾选;③确认模型选择并从数据库中获取模型参数及相关配置。

(3) 模型组合: Web 服务具有组合的功能,即服务组合,通过服务组合可以实现更为复杂的服务,将每一个原子模型封装成一个服务,在执行复杂模型时,实际上就是执行多个原子服务的一种特定组合,通过服务组合实现不同分辨率的模型聚合,最终将组合后的模型进行统一封装,实现模型集成应用。

(4) 模型组装:主要是针对组件模型的装配,组装是通过系统化构建方法实现模型功能的具体方式,以组件和各种元模型为基本要素,通过模型框架快速配置生成组合模型,并通过组件装配模型与代码模板自动生成部分模型执行程序。因此,设计一种用于模型组装的连接件模型,如图 6-5 所示,主要由以下几个部分组成。①连接件描述。连接件按照兵种专业进行划分,如炮兵连接件、防空兵连接件、坦克连接件和通信兵连接件等。可以按照组装的模型组件类型选择不同的连接件进行装配,其描述主要包括名称、类型、功能说明等内容。②连接件方式。作为模型组件连接的纽带,连接方式主要包括顺序、循环、并行 3 种。③连接件接口。指的是组件之间连接的接口,包括各个相互连接组件的名称、类型、接口规格、具体功能等。④组件装配执行顺序。根据连接件类型设定组件之间的执行顺序,主要是根据条件选择执行顺序。⑤连接实现。包括实现技术类型、引用位置、内容、说明等内容。⑥连接件属性。包括所连接的组件的名称、类型、功能等内容。⑦数据处理。主要完成模型组件内部和外部数据转换、交互和传输等。⑧服务接口。指的是通过连接件对组件进行装配后形成的对外提供服务的专用接口,通过对此接口的描述可实现模型服务,主要是对服务接口信息的具体描述,如名称、类型、接口规格和功能等。

采用 XML 模式对连接件模型进行描述,具体如下:

- (1) 分别将连接件模型的 8 个部分描述为 XML 模式的 Element。
- (2) 分别将每个组成部分的具体内容描述为 Attribute。
- (3) 分别为 Element 与 Attribute 定义简单或复杂类型。

模型组装就是将需要组装的模型组件的多个服务接口绑定到组装后的组件

服务接口上，并建立起多个服务接口之间的连接关系和交互方式，通过连接件实现不同组件之间的协调运行和访问。连接件是组件装配的基础，模型组件组装机制是连接件完成各模型组件交互的具体体现。

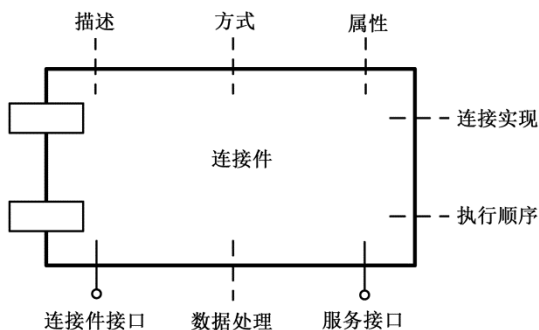


图 6-5 连接件模型

6.3.4 应用层

仿真模型集成框架的应用层构筑在服务层上，针对不同的应用，提供仿真模型服务，用户只要针对某一应用，输入相应需求参数，无须考虑模型调用模型资源、完成模型组合与运行的具体过程，使用户能够方便、快速地实现仿真模型应用。

应用层主要针对模型定制应用、模型测试应用、模型验证应用和作战仿真应用等，通过统一的仿真模型服务访问接口实现模型的集成应用，主要包括模型访问、模型控制、模型交互、模型调度和模型运行等，在模型组合的基础上进行集成应用。

通过提供友好的输入界面，按照模型服务的应用领域和功能，帮助用户提交各种模型应用请求。用户启动应用程序，通过 API 向服务层提出模型服务请求，服务层将任务请求解析后，提取仿真模型应用的语义信息，通过模型调度管理器调用模型服务，实现模型服务实例的封装，驱动模型资源完成任务，基于 Web 服务的模型组合应用如图 6-6 所示。

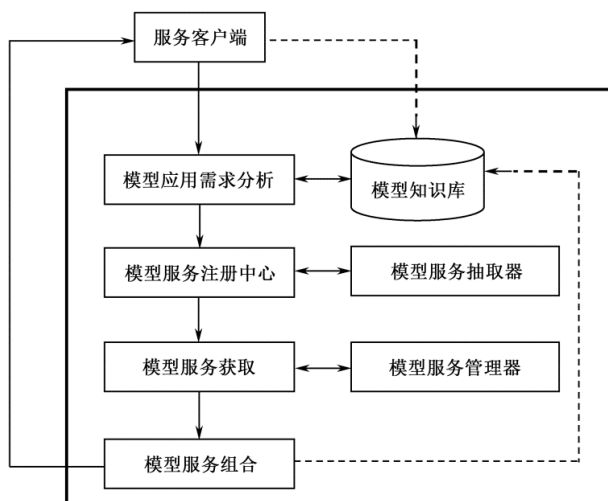


图 6-6 基于 Web 服务的模型组合应用

1. 服务客户端

模型服务客户端提交模型服务应用请求，通过提供相应的主题词表和导航目录，在应用领域和服务功能两个层次上对用户的模型服务请求进行限定，减少模糊性和歧义性，同时为在一定范围内选取可用模型服务创造条件，提高仿真模型服务搜索和获取效率。

2. 模型应用需求分析

根据用户的模型请求信息得到所需模型服务的初步描述，然后通过模型知识库中的领域信息确定相应领域，综合分析已有的用户兴趣模型和组合方案，根据用户所需的模型服务，在该领域中选定包含这些服务功能的组合方案。

3. 模型知识库

模型知识库用来存储仿真领域知识、用户兴趣模型和组装方案。Web 客户端根据用户服务管理中的领域信息向用户提供领域导航，同时还将用户兴趣模型及行为记录在知识库中，以便下次向用户提供更为方便的查询。在领域知识中，服务组合方案为用户的模型服务需求提供了原始方案，所有的模型组合方案都是由这些原始方案生成的。如果模型服务组装验证可行，并使模型服务成功完成，产生的组合方案就会存储到模型知识库中以备以后复用。

4. 模型服务注册中心

模型服务注册中心存储并定时更新各种模型服务信息,并发布已经注册的模型服务,使服务请求者可以通过模型服务注册中心查找特定的仿真模型服务,通过模型服务抽取器实现对所有相关模型服务信息的抽取。

5. 模型服务管理器

主要完成对模型服务的发现、查询、修改和管理等,管理的主要是各种仿真模型服务的描述信息、服务接口等。它通过返回的模型服务消息来响应服务查询并与服务提供者交互,由服务提供者来创建满足需求的服务实例,并返回这些服务实例的 SH (Service Handle) 和初始的 SR (Service Reference)。

6. 模型服务组合

按照模型服务组装方案定义的服务调用关系,对选定的模型服务进行连接和组合。在组合过程中,根据用户的具体需求调整和设置各个服务的可定制属性,完成用户所要求的模型服务功能。

6.4 仿真模型集成框架功能设计

基于 SOA 架构模式,采用面向服务的设计方式,按照组件化建模规范,通过本体映射的方式实现模型映射,基于模型驱动架构完成模型转换,重点研究作战仿真模型开放式集成框架。该框架具有模块化、可复用、可扩展和流程控制等多种特性,能够适应陆军部队作战训练与作战实验仿真模型的快速开发和灵活构建,具有横向联合各军兵种、纵向实现多作战层次的仿真模型应用集成,满足军事训练、武器装备论证、战法研究等多种军事仿真应用需求。从功能和处理流程上,该框架主要具备模型管理、模型封装、模型组合和模型集成应用等功能。

6.4.1 模型管理功能

为了控制建模过程的复杂性,需要对模型进行有效管理。模型管理就是通过统一的框架对模型资源进行组织、存储及运行过程的全方位管理,以实现快速重

构的目的。通过提供统一完备的模型管理机制，使模型库具备可扩充、可修改、易维护等特点，便于模型的灵活访问、更新、生成和运行。

因此，模型管理的难点在于^[66]：一是模型如何分解存储到模型库；二是模型如何运行；三是模型如何组合使用。一方面要关注如何根据用户的需求查找满足需求的模型服务并形成模型组合方案；另一方面要关注仿真的柔性、鲁棒性和运行效率，主要是模型的动态重构和组合等问题。

对仿真模型管理通常需要考虑几个问题^[67]：①模型重构快速性；②模型分类存储的合理性；③模型管理的动态性和灵活性；④模型调度的高效性；⑤模型运行控制的有效性；⑥模型组合的可靠性。

模型管理包括模型存储管理、模型更新管理、模型访问管理和模型运行管理。主要提供模型的维护操作，包括对模型库进行打开、创建、删除、更新和注册等；提供模型的存储管理，包括模型的表示、存储的组织结构和查询；提供模型控制功能，包括模型的选择、运行与结果显示、集成和组合。

模型管理系统以文件形式存储和管理模型，并与数据库实时交换数据，将模型与其相关的数据分开进行管理，有利于充分发挥文件系统的可变大容量文件存储优势和数据库的复杂关系数据管理优势。模型管理主要由建模工具、模型配置器、模型控制器等组成，如图6-7所示。模型配置器主要是完成模型的各种配置文件和参数，实现对模型的注册和更新管理。模型控制器主要完成模型的各种交互功能，实现模型的访问和运行。模型管理系统与数据库系统共同组成模型资源管理系统，通过模型资源管理系统实现对模型和数据的集中、统一访问^[68]。

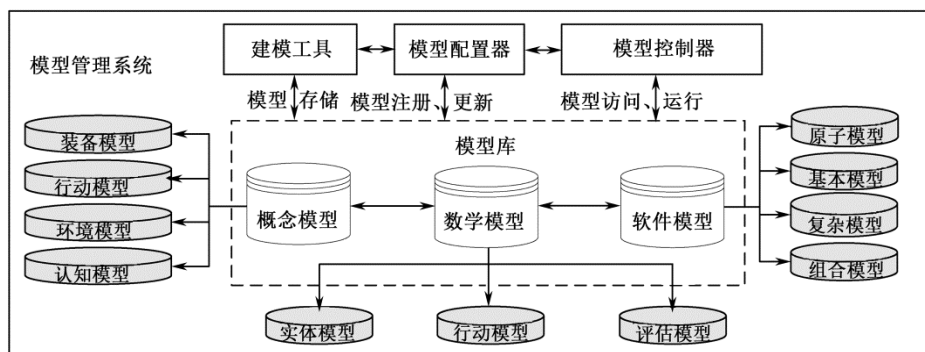


图 6-7 模型管理

模型库主要用于存储各类模型，是系统的核心部分。在逻辑上，模型库是各

种模型的集合；在软件内容上，则由许多计算机内的程序模型组成。模型库、数据库与管理信息系统之间的逻辑关系和结构如图 6-8 所示。

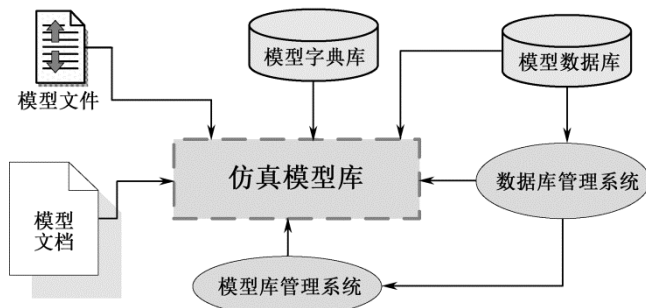


图 6-8 作战仿真模型库系统结构

(1) 模型字典库：是基于关系数据库的组织结构形式，在数据库的表空间中存储模型和相关文档的标识信息、模型名称、模型类型、存储位置、模型编写者、编写时间、功能参数、与模型间的关系等信息，以便对模型操作。模型字典库的组成如图 6-9 所示。

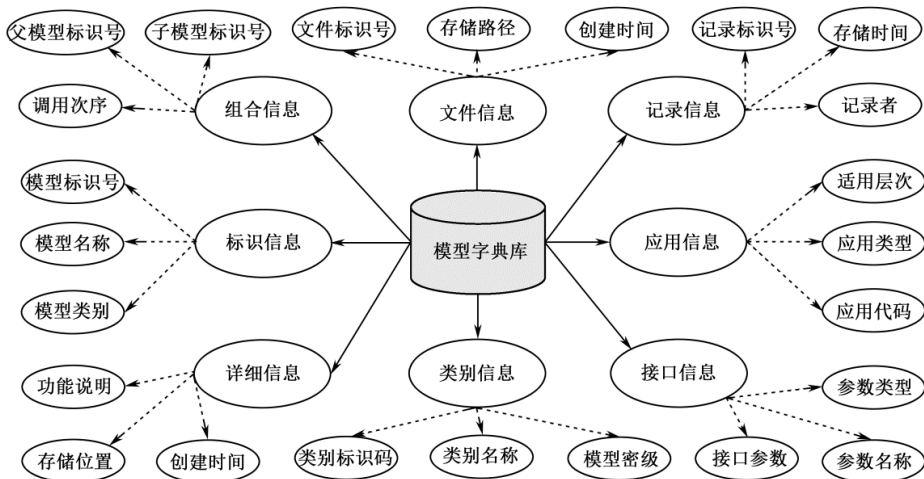


图 6-9 模型字典库的组成

模型字典库一般包括以下内容：

- 模型标识信息：模型标识码、模型名称、模型类型、密级等；
- 模型详细信息：模型功能简介、存储位置、创建时间、修改时间、编写者等；
- 模型类别信息：模型类别标识码、类别名称等；
- 模型调用接口信息：接口参数类型、参数名称、源接口、目标接口等；
- 模型组合关系信息：父模型标识码、子模型标识码、调用次序等；
- 模型文件信息：模型文件标识、文件存储路径、创建时间、修改时间等；
- 模型结果记录信息：提供模型计算结果存储记录的所有信息，如：记录标识码、记录存储时间等。

(2) 模型文件：指模型的源代码或目标代码（动态链接库），便于模型调用；

(3) 模型文档：指模型的相关说明文档，方便用户通过系统查看模型功能的详细描述信息。

模型库系统是以文件形式存储和管理模型的，所以，模型管理系统实际上同时管理了相关的模型库和数据库，实现了模型与数据的分离。模型库中的模型和相关的数据库中的数据都受模型库管理系统的管理^[69]，只有通过模型库管理系统才能够访问模型和相关的数据库。模型库管理系统的总线结构如图 6-10 所示。

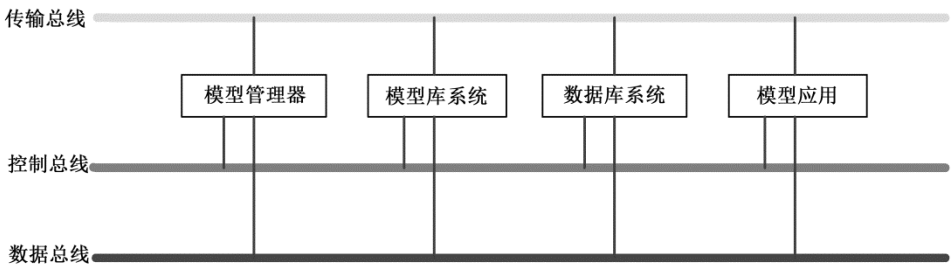


图 6-10 模型库管理系统的总线结构

对于模型的描述采用基于元数据和 XML 的方法，模型的元数据（Meta Data）是关于数据的数据，它是关于模型对象的、有应用需求的属性数据或信息，用于描述、解释和说明模型的主要性质和特征，以便对模型进行操作，如组织管理、查询搜索、获取传输、共享等。模型库系统中关于模型的描述、存储、组合和使用等信息称为模型的元数据。模型的元数据不仅能指明模型的作用，定义模型信

息的内容和位置,表述模型的抽取和转化规则,描述模型的主题和相关信息,而且还能实现模型的管理(例如,修改和跟踪模型信息),描述模型同步需求,衡量模型质量等功能。模型元数据代表了整个模型库系统的逻辑结构,成为整个分布式模型库系统的管理核心。

采用目录树的方式实现对模型的层次化管理,将所有模型按一定类型分类管理和显示,创建一个模型目录树,需要在模型字典库中将每个模型按照树型结构进行层次分类编号,通过数字序列的方式对各子模型进行分类,便于模型的识别和调用,如图 6-11 所示。利用 LDAP (Lightweight Directory Access Protocol) 目录服务器存储模型库共享信息资源,后台采用 Mysql 数据库支持,并应用 Web 服务作为统一访问接口,满足用户对模型库的透明性访问,实现对模型库的分布式管理。按照树形结构组织成为一棵目录信息树 DIT (Directory Information Tree)。DIT 中各目录由条目(Entry)组成,根结点到每个目录项的路径称为 DN (Distinguished Name)。条目是具有 DN 的属性(Attribute)集合,属性由类型(Type)和多个值(Values)组成。

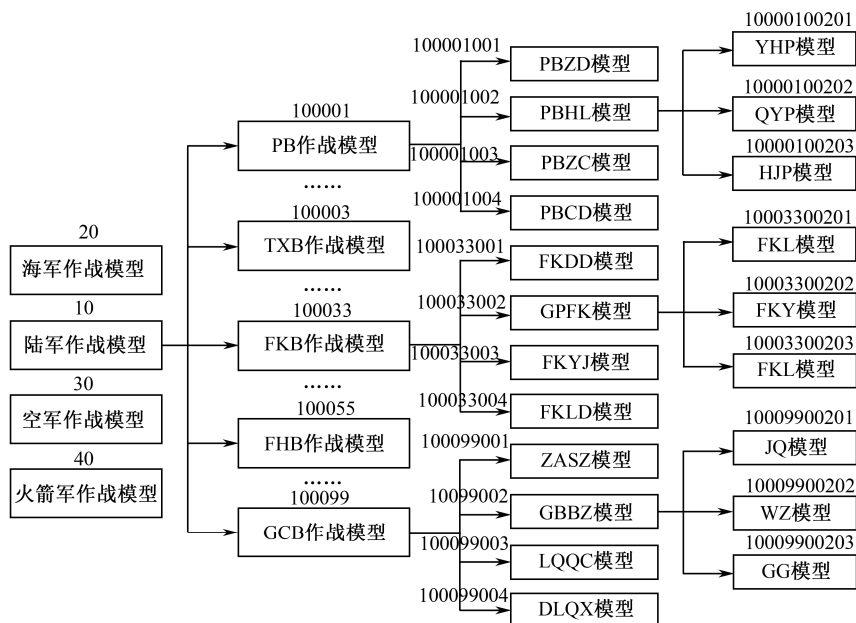


图 6-11 模型资源目录树

LDAP 特别适用于分布式模型资源管理,符合分布式信息处理模式,具有很好的层次性和扩展性,在对模型库进行资源管理的访问查询时能够避免集中式信息检索带来的瓶颈等局限性,可以方便模型调用。

6.4.2 模型封装功能

对模型的封装实际上需要经过解构、拆分和封装三个过程,主要涉及下面三个问题:①解构问题。按照多大的粒度对模型进行解构,使得所有被解构的模型以统一的标准进行粒度区分和存储。②接口定义问题。经过解构和拆分后,需要按照统一的规范对模型接口进行重新定义,便于模型在组合、集成应用阶段的交互,提高可重用性。③封装问题。按照组件化建模的要求,需要对模型进行统一封装和标识。

通过制定开放性、可扩展的模型封装规则、流程,设计模型识别器,完成对模型的自动标识,主要完成模型描述、模型配置、模型重构和模型装配等。

模型在封装之前,通常需要将组件模型按照统一的规范进行转换,模型转换的方式有三种:一是技术人员利用自己的专业知识,完成对模型设计说明的人工转化,然后编码实现;二是把模型框架与共有功能提取出来,形成模型开发向导,技术人员利用模型开发向导,实现模型半自动转化;三是充分利用前两种方法,定义模型间协议接口规范,把技术人员对模型的推理、转化与实现过程进一步抽象,描述成可以自动推理的“推理机”,实现军事概念模型到程序模型的自动转化。这里采用第二、三种模型转换方式,参照 COM 组件对象的软件工程方法,将模型分成原子模型、基本模型、复杂模型和组合模型等几个层次,并以服务的形式对其进行封装,针对不同的模型实现不同粒度的服务组合,将原子模型、基本模型、复杂模型和组合模型分别以原子服务、基本服务、复杂服务和组合服务的方式建模,完成仿真模型组件开发。首先,依据作战实体需求模型,完成模型总体设计,其基本原则是使得每个模型尽量独立,能够通过其他模型生成的实体模型不单独建模,尽量减少模型间的重复;其次,将不同的实体模型进行服务封装;再次,设计服务组合的规则,使得各种作战仿真模型能够以服务的形式灵活组合;最后,完成对仿真模型服务组件的测试和管理。以指挥实体模型组

件封装为例,图 6-12 是具体的过程,其中编程与封装主要是由服务抽象、功能编程、服务封装和服务组合策略等组成。模型组件中包含通用组件和专用组件,这种组件实际上就是通过统一、规范的服务接口实现通用服务、专用服务和组合服务等。

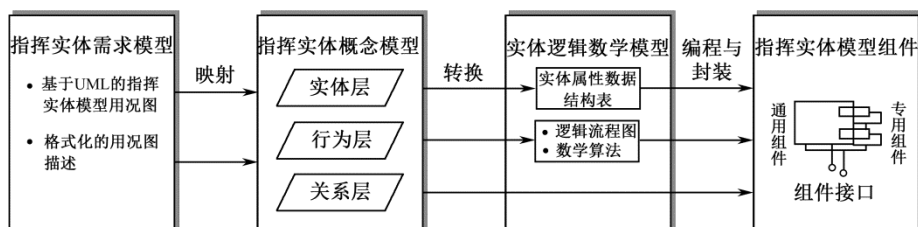


图 6-12 模型组件封装

可以看出,作战实体一体化建模首先根据军事模型需求分析,按照规范化建模方法,建立不同形态、不同功能的军事概念模型和数学模型,然后根据预定义的模型协议接口规范,在统一集成框架支持下,采用服务封装的方式实现不同形态模型间有机衔接,最终生成可重用、可集成的作战实体模型组件服务。

从 Web 服务技术的实现机理和过程可以看出,采用 Web 服务技术,实现作战仿真模型的组件化、规范化,是促进建模标准化,提高作战仿真模型可重用性和互操作性的有效途径^[70]。主要优势有:①仿真模型资源充分共享,提高资源利用率;②兼容性好、适用性强,不受操作系统和应用软件的约束和限制;③具有良好的封装特性,使得模型内部机制得以安全、完整维持,避免模型被任意改动;④访问接口统一、标准化,所有的应用均以服务方式呈现,易于访问。基于服务封装过程,通过服务的组合实现模型的集成调用,以达到模型重用目的。模型功能组件以服务组合形式出现,按照一定的策略实现服务组合,并通过实例化完成相关作战仿真活动,将软件构架、通用服务组件、专用服务组件加载到作战仿真系统中。模型组件的装配过程如图 6-13 所示,将每个独立的组件模型进行接口关联、参数配置和装载,再通过专门的模型测试系统进行验证和测试。

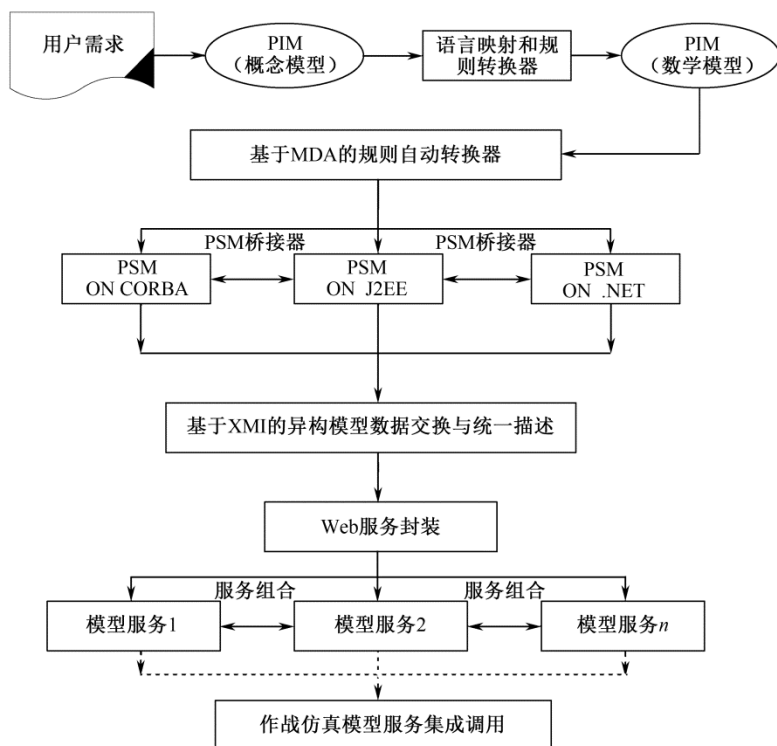


图 6-14 模型组件服务封装

6.4.3 模型组合功能

组合建模可以快速构建新模型，提高模型重用性。模型组合是一个层次化管理过程，一个复杂模型可以通过一系列原子组件模型组合而成。原子组件模型具有标准输入、输出端口，按照标准协议完成信息传递。通过不同粒度的组件模型构造更高层次的复杂模型是进行模型组合的主要目标。

通过建立多粒度仿真模型组合、控制机制，使其能够根据不同模型的领域视角、应用需求描述，实现多种仿真模型在同一作战背景下的联合运行、聚合与解聚。将模型组合以服务组合的方式来实现^[72]，将组件模型封装成服务后，以逻辑规则表示领域知识对组合模型的约束和接口匹配关系的约束，建立相应的模型组合规则、约束规则、消息传递规则和接口匹配规则，并根据领域规则和逻辑推

理引擎实现对组合模型的约束检验和接口匹配检查。主要完成模型发现、模型适配、组合分析、组合配置和模型组装等功能。

原子服务是一种不可再分的服务,组合服务是由原子服务构成的具有复杂功能的服务。首先由用户提出申请需求服务,即对所需模型服务进行描述,根据模型接口描述和功能描述,进行语义提取与转换。语义提取是与模型知识库进行比对的过程,并将提取的语义转换成组合推理引擎能够识别的语言。在组合规则库和模型知识库的共同作用下^[73],推理引擎按照模型组合服务流程对模型进行组合封装,然后将组合结果反馈,并进行模型组合性验证,最后将组合结果以服务的方式提供给用户直接使用。组合服务的基本过程如图 6-15 所示。

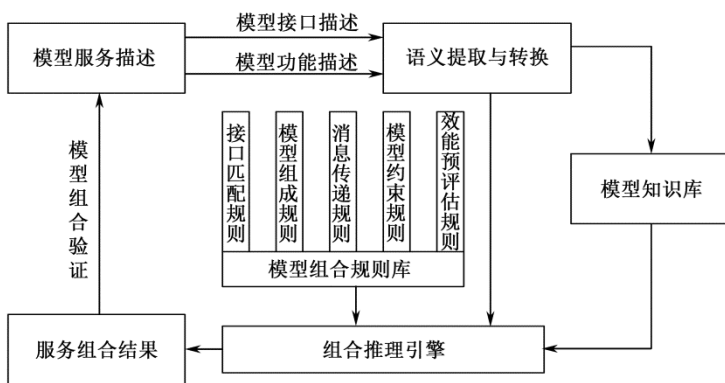


图 6-15 组合服务基本过程

设 ws_1 和 ws_2 是两个原子服务,原子服务只有一种结构,如图 6-16 所示。而组合服务有三种结构,分别是顺序结构、并列结构和循环结构,对应采用算子 \oplus, \otimes, \odot 表示。

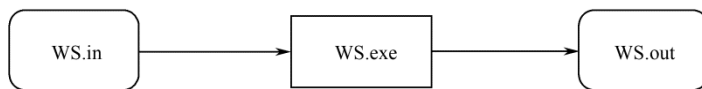


图 6-16 原子服务模型

(1) 顺序组合服务: 是一种按照先后顺序进行组合运行的服务形式,如图 6-17 所示,表示为 $WS = ws_1 \oplus ws_2$ 。

(2) 并列组合服务: 是一种可以同时进行的服务运行形式,如图 6-18 所示,表示为 $WS = ws_1 \otimes ws_2$ 。

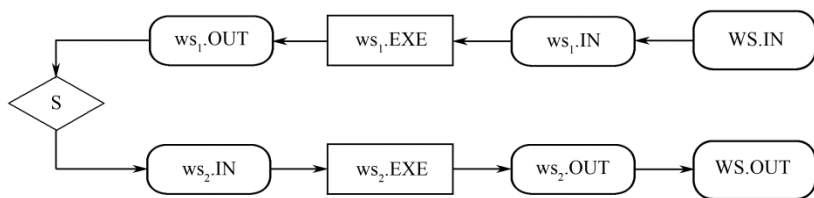


图 6-17 顺序组合服务模型

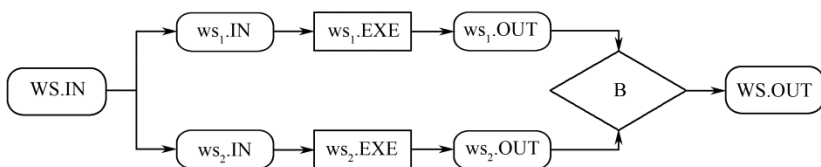


图 6-18 并列组合服务模型

(3) 循环组合服务：是一种可以多次重复进行的服务运行形式，如图 6-19 所示，表示为 $WS = ws_1 \odot ws_1$ 。

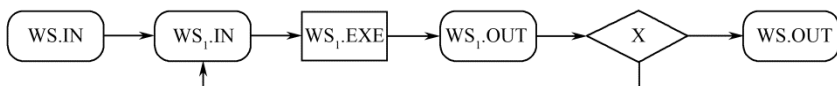


图 6-19 循环组合服务模型

假设要完成一次防空火力打击任务，考虑到地理位置、我方防空力量打击效能、敌方攻击武器性能、防御时间和作战单元战斗力综合水平等因素，只能在 A、B 两个防空单元进行选择，A 单元距离来袭目标较近，所属人员战斗精神较好，且 A 单元具有较好的侦察设备，B 单元距离来袭目标稍远，但武器攻击效能较好，且具有较好的抗干扰能力，但由于 B 处的气象条件相对较差，阴天且风力较强，还有一些其他的影响因素，需要进行综合效能打击评估后才能最终选择作战单元。将此项决策活动看作是一个组合服务 WS，共有 4 个基本的服务：对作战单元 A 进行评估的服务 ws_1 ，对作战单元 B 进行评估的服务 ws_2 ，对作战单元 A、B 进行效能排序比较的服务 ws_3 ，选择 A 或 B 进行打击的决策活动，即作战决策的服务 ws_4 。

通过进一步分析可知， ws_1 和 ws_2 是两个并发逻辑的服务，几乎可以同时进行，而 ws_3 和 ws_4 是两个顺序逻辑服务，因此，此项服务组合可以描述为

$WS = (ws_1 \otimes ws_2) \oplus ws_3 \oplus ws_4$ ，其结构如图 6-20 所示。

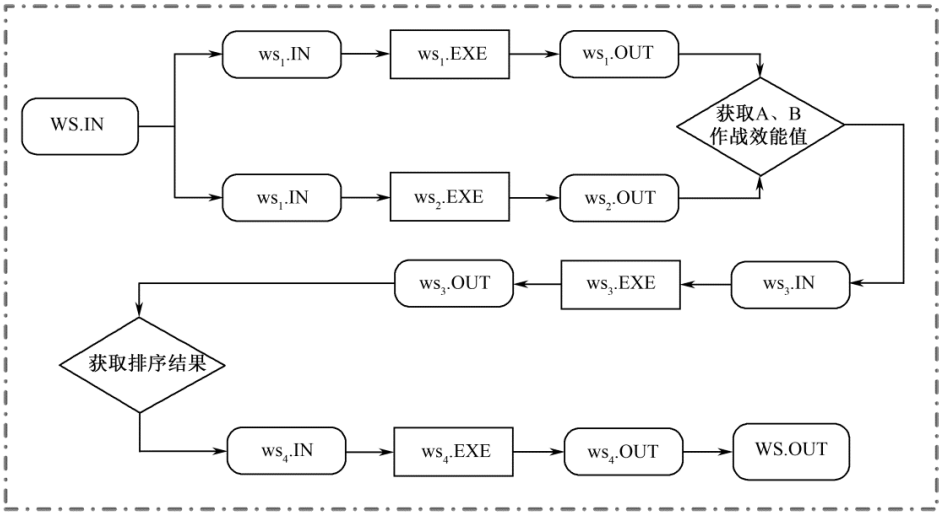


图 6-20 实例化的服务组合流程

从视图的角度，将仿真服务组合的实施过程分为需求视图、模型视图、服务视图和运行视图，如图 6-21 所示。

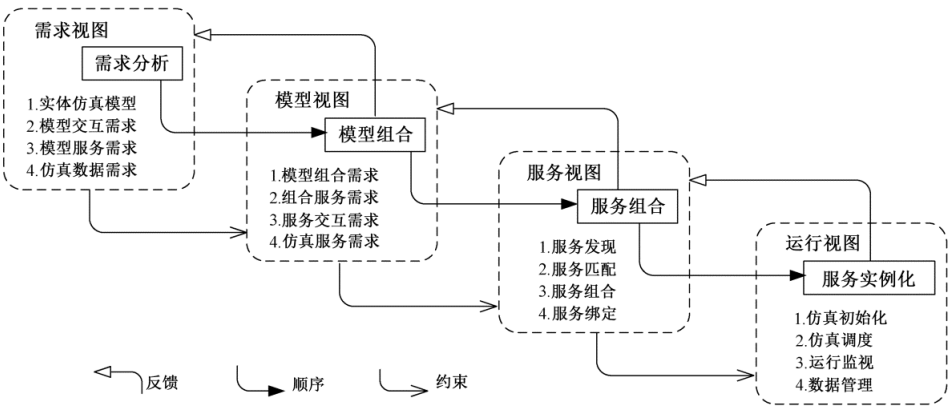


图 6-21 模型组合实施过程

需求视图主要根据仿真应用需求确定实体仿真模型、模型交互关系、模型服务需求、模型与服务之间的逻辑关系及仿真应用过程中的数据需求。

模型视图指组合服务模型的设计视图,是进行模型组合的基础,需进一步细化模型组合需求、组合服务需求、服务交互需求和仿真服务需求,根据模型视图确定组合服务模型。

服务视图主要是完成服务发现、服务匹配、服务组合和服务绑定等,基于 SOA 的集成框架根据组合服务模型中定义的仿真服务需求查询服务代理,进行服务匹配,发现满足功能和非功能需求的候选仿真服务集合^[74]。根据特定的服务选择策略选择服务,通过相关映射建立组合方案,并将满足要求的服务与用户应用进行绑定。

运行视图是服务实例化的过程,主要完成仿真初始化、仿真调度、运行监视和数据管理等,首先依据组合方案创建组合实例,并根据交互需求进行仿真调度,协调组合模型之间的数据交互。

6.4.4 模型集成应用功能

目前,模型集成的研究主要集中在三个方面:①模型表达;②模型的集成方式;③模型的集成框架和支撑环境。军事需求的增长促进了作战模型的增加,原有的和新开发的模型混在一起,增加了模型运用的复杂度。为此,需要把建模、管理、使用作为一个大的总体系统来考虑,使模型的开发、管理、调用在统一的框架、模式下进行,对模型开发规范、集成应用进行一体化、规范化研究。

根据仿真应用需要,描述各类功能模型构件加入或退出集成框架的条件及实施方法。根据作战仿真模型集成框架运行的需要,描述各模型在集成框架运行中的控制机制。根据不同的应用需求主要完成模型访问、模型控制、模型交互、模型调度和模型运行等功能^[75]。总的来说,模型集成应用服务模式分为模型下载式应用服务模式和模型发布/订阅式应用服务模式。

下载式服务模式主要是根据用户的请求,在模型名录中进行搜索,然后在搜索的结果中进行模型定位^[76],根据相关定位信息从模型库中获取模型文件(主要是程序文件或功能函数模块),以下载的方式将模型保存至本地应用,或是直接嵌入到系统中,在本地完成编译后运行仿真模型,其具体流程如图 6-22 所示,这种方式主要查询获取,没有智能推理和分析。

在分布式系统中，以信息分发技术为基础，采用以信息分发为核心的发布/订阅式服务模式，如图 6-23 所示。这种共享方式使发布者或请求者的界定相对模糊，发布者可以发布模型，也可以申请服务，同样申请者也一样。基于发布/订阅机制可以构建一个多对多的访问模式，摆脱用户对时间和空间的束缚^[77]。发布与订阅的节点没有严格的划分，发布者按照规范进行模型发布，经过 VV&A 后将模型注册后入库，并生成相应的模型文件。申请者根据实际需求，提出模型申请需求，经过组合分析后，在模型库中提取相关组件模型，并按照一定的逻辑顺序和装配关系进行组合，在此基础上进行服务封装，然后通过绑定的 URL 向用户提供模型服务。

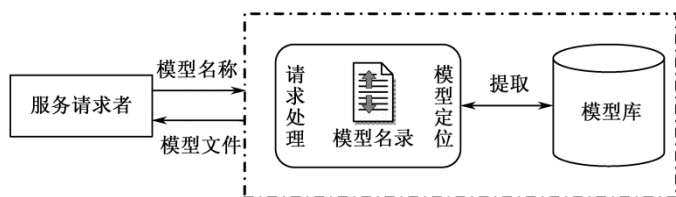


图 6-22 下载式服务模式

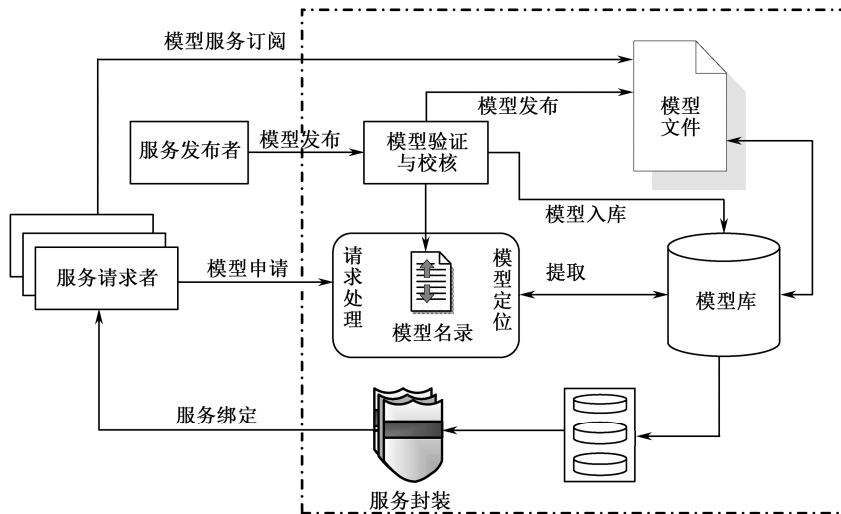


图 6-23 发布/订阅式服务模式

6.5 本章小结

本章首先对现有模型集成框架存在的问题进行分析,结合当前作战仿真的实际需求,提出了一种面向服务的模型集成框架,构建基于 SOA 的陆军作战仿真模型柔性集成系统,实现对模型构件库、建模知识库和模型数据库的统一组织与管理。参照面向服务的体系结构,采用分层描述的方法,设计了四层仿真模型集成框架:资源层、构件层、服务层和应用层。采用服务的方式实现模型的集成应用,通过平台无关的通用协议进行服务调用,按照服务规范对仿真模型服务进行描述并发布,通过服务发现和服务选择来动态构建满足实际需求的更为复杂的组合服务,以满足不同需求的仿真应用,实现模型管理、模型封装、模型组合和集成应用等功能。

第 7 章

总结与展望



7.1 研究工作总结

本书中，首先，阐述了作战仿真模型体系相关问题研究意义和思路，研究了模型体系的形式化建模方法。然后，按照组件化建模思想完成对陆军仿真模型设计并构建了可组合的仿真模型体系。在此基础上，通过服务组合的方式实现模型组合，设计了一种面向服务的模型集成框架，并对框架的功能进行了详细描述与设计。其主要工作包括以下方面：

(1) 采用解释结构模型 (ISM) 方法对模型体系中模型间的继承关系和组合关系进行建模，分析了各个模型之间的关系，将模型体系构造成一个多级递阶的结构模型，对模型间的继承关系、组合关系进行了形式化分析，为构建可组合的仿真模型体系奠定理论基础。

(2) 提出了一种基于模板的军事概念模型建模方法 MCMT (Military Concept Model based on Template)，主要从原子角度对概念模型进行划分，通过实体元模型、动作元模型、关系元模型、属性元模型和数据元模型等对军事概念模型进行描述。在此基础上，采用组件化建模思想，设计了一种新的组件化建模流程，从新的视角对模型组件进行划分，构建了一种由组件库、对象库、基本库和单元库

构成的模型组件架构,采用“基础+维度”的方式设计了一种可组合、可扩展的仿真模型体系,通过映射、关联、组合和聚合关系实现模型的快速重构、装配和组合。为提高模型组件的通用性,将模型组件接口设计分为组件描述接口、组件组合接口、组件功能接口、输入和输出接口,解决了模型的兼容性问题、跨平台重用问题和异构模型交互问题。

(3) 为简化模型组合的实现方式,同时又能满足多样化作战仿真对模型服务组合的需求,针对作战仿真模型组合规则复杂、基本服务类型相对单一等特点,提出了一种基于事件的服务组合 BEWSC (Based-Events Web Services Composition) 方法。首先定义一种基于 MERA (Mission-Event-Restriction-Action) 规则化语言,在此基础上,采用过程构造子,按照模块化方法对模型服务的组合方案进行描述,并分别从服务的发布者、服务的请求者和推理过程这三个角度对组合服务的执行流程进行具体分析,解决了服务组合域表示困难、描述能力不足、模型匹配差异性大和模型组合有效性等方面问题。

(4) 参照面向服务的体系结构,采用分层描述的方法,提出并设计了一种基于 SOA 的仿真模型集成框架 IFSM-SOA (Integrated Framework of Simulation Model based on Service-Oriented Architecture)。在该集成框架下,基于 MDA 方法实现模型的统一转换,采用组件化技术实现对模型的重构、装配,并以服务的方式实现模型封装、组合和调用等,使不同类型、不同层次和不同粒度的模型以一种通用、统一的方式进行交互和共享。

7.2 需进一步深入研究的问题

在本书中,只是以陆军部队为基本研究对象,对仿真模型体系涉及的一些关键问题有比较深入的研究和一些创新,但由于个人能力、精力和实验室条件等方面的限制,面对复杂的模型体系,还有很多问题考虑不全面、研究不透彻,书中以陆军仿真模型为研究对象,实际上只是在一定范围内进行尝试,希望能起到抛砖引玉的作用,为在更大范围内研究模型体系的关键问题提供一些思路和借鉴。还有很多内容值得进一步深入研究和探讨,主要包括以下几个方面。

1. 模型体系的理论研究

在大规模作战仿真系统中,模型体系的构建是十分重要的,特别是对于联合作战仿真而言,模型种类会更加繁多,模型之间的关系会更加复杂,如果没有科学、合理的模型体系作为支撑,对于模型的运用存在很多的局限。只是以陆军作战仿真模型体系为例,在一定范围内对其模型体系的结构进行了形式化建模与分析,理论基础的成熟性还需要进一步探讨。下一步有必要结合体系工程、复杂系统和形式化建模的理论和方法加强对模型体系的研究,加深对模型体系的认识和理解。

2. 模型组合研究

采用服务组合的方式实现对模型组合的研究,组合语法、规则和算法还不够完善,为了实现更大规模的复杂模型服务组合,还需要对模型服务在语义层进行描述和推理,解决带有选择、分支和汇合等复杂流程关系的并发服务组合问题。另外,服务组合规则对于组合的可靠性起着重要的作用,随着模型组合需求的不断增加,组合规则需要不断更新和完善,以满足复杂仿真模型的组合需求。同时,本书只研究了一种组合算法,而实际上,要想解决并获得良好的组合方案,一种组合算法是难以完成的,需要针对不同类型的组合应用采用不同的组合算法。

3. 组件化建模研究

本书对组件化建模的方法进行了研究,对模型开发和设计有一定的指导作用,但离工程化实践还有一定的差距,还需要进一步研究组件化建模的标准并开发通用建模工具,通过图形化界面自动完成组件接口的定义和相关参数的配置,并根据不同类型的特点,提供不同的操作界面,规范建模过程,通过固定的界面消除不同建模人员的歧义性,使得建模人员能够按照固有的组件开发过程实现对模型组件接口设计和参数设置,为组件组装奠定统一访问的基础。

4. 模型集成框架研究

模型集成框架能为模型建立统一的集成机制和模型交互的统一运行环境,当前典型的模型体系框架主要有 FLAMES、JSIMS、OneSAF 和 XMSF 等,书中提出的一种面向服务的模型集成框架,只是完成了框架的总体设计,从资源层、构件层、服务层和应用层对框架进行了设计,并对框架应具备的模型管理、

封装、组合和集成应用等功能进行了描述，仅通过一个原型系统进行了初步的测试与实现，对于大规模的仿真应用支持程度还需要进一步论证，特别是对于多用户的分布式应用，还需要研究如调度策略、模型监控管理和计算资源合理分配方式等问题。

参 考 文 献

- [1] Temizer S. The State of the Art and the Future of Modeling and Simulation Systems [J]. Journal of Aeronautics and Space Technologies, 2007, 3 (1):41-50.
- [2] Hao XiaXia .College of Information Science and Technology[J]. Nanjing University of Aeronautics and Astronautics. Overall Framework Design of Simulation Operation Model Base System And its Realization,2008.
- [3] 冯军. 模型库管理系统的开发研究[J]. 兵工自动化, 2010 (05) .
- [4] 陈欣. 美军建模仿真对象模型体系框架研究[M]. 北京: 军事科学出版社, 2008.
- [5] 张琦. 使命空间功能描述理论和方法研究[D]. 长沙: 国防科技大学研究生院, 2005.
- [6] 马亚平, 李柯, 崔同生. 联合作战模拟中武器装备体系结构研究[J]. 计算机仿真, 2004, 21(3): 7-9.
- [7] 金伟新, 肖田元, 马亚平. 联合作战仿真模型体系的设计[J]. 计算机仿真, 2003, 20(8): 4-6.
- [8] 丁红勇. 空间系统军事应用仿真模型体系设计[J]. 系统仿真学报, 2007, 19(18): 4292-4294.
- [9] 彭英武, 李庆民, 王睿. 面向任务的装备保障仿真模型体系研究与应用[J]. 海军工程大学学报, 2010, 22(5): 62-68.
- [10] 王鸿洁, 常国岑, 李学军. 军事通用仿真模型及其集成框架研究[J]. 系统仿真学报, 2007, 19(3): 486-490.
- [11] 曾艳丽. 电子战仿真模型构件库的研究与设计[J]. 中国电子科学研究院学报, 2006 (02) .
- [12] 张震. 作战仿真模型库系统研究[J]. 计算机工程, 2004 (13) .
- [13] 吴泽彬, 吴慧中, 李蔚清, 等. 面向分布式仿真资源库的统一检索研究[J]. 兵工学报, 2008 (01) .
- [14] 程思微. 基于代理的航天装备资源管理系统研究[D]. 长沙: 国防科学技术大学, 2004 .

- [15] 顾冬娟, 夏磊. 基于决策支持系统的模型库系统研究[J]. 硅谷, 2008 (12).
- [16] 郑颖华, 武根友. 智能决策支持系统中的模型库及其管理系统[J]. 科学技术与工程, 2006 (09).
- [17] 黄莺, 郭煌, 惠晓滨, 等. 基于面向对象的智能决策支持系统模型库自动选择[J]. 兵工自动化, 2007 (03).
- [18] 陈得宇. 指挥决策支持系统中模型库的研究[D]. 哈尔滨: 哈尔滨工程大学, 2005.
- [19] 赛英, 董宁, 聂培尧. 面向对象的模型库与数据库接口技术[J]. 计算机工程与设计, 2007(01).
- [20] 董晶晶, 张利红, 陈兵, 等. 浅谈利用面向对象的方法实现 GIS 模型库系统[J]. 地理空间信息, 2010(06).
- [21] 雷永林. 仿真模型重用理论、方法与异构集成技术研究[D]. 长沙: 国防科学技术大学, 2006.
- [22] 杨明, 张斌, 王子才. 建模与仿真技术发展趋势分析[J]. 系统仿真学报, 2004(9).
- [23] 彭勇. 基于领域规则的军用仿真模型组合建模[J]. 系统仿真学报, 2011,23 (8): 1527-1532.
- [24] 孙万国, 王学智, 杜峰. 美陆军数字化部队装备保障特点及其启示[J]. 装甲兵工程学院学报, 2010,24(6): 17-19.
- [25] 王平军. 未来数字化战场信息技术发展探析[J]. 舰船电子工程, 2003,4(2): 11-14.
- [26] 邹逢兴, 李国川, 谢石岩. 数字化战场与战场数字化技术[J]. 国防科技大学学报, 2008,20(6): 51-54.
- [27] 胡晓峰. 作战模拟术语导读[M]. 北京: 国防大学出版社, 2004: 21-24.
- [28] 叶丰, 蔡业泉, 陈欣. 武器装备体系对抗仿真模型开发关键技术研究[J]. 军事运筹与系统工程, 2008,12(22).
- [29] 许金裕. 数字化部队论[M]. 北京: 解放军出版社, 2004:15-20.
- [30] 彭勇. 作战仿真模型体系分析及其模型设计与实现关键技术研究[J]. 国防科学技术大学研究生院, 2011,10:57-68.
- [31] 周德群, 方志耕, 潘东旭. 系统工程概论[M]. 北京: 科学出版社, 2007:117-137.

- [32] 刘淑芬, 赵金红, 许鹏. 武器装备仿真模型体系框架的研究及应用[J]. 吉林大学学报(理学版), 2010,6(48): 1008-1012.
- [33] 燕雪峰, 杜庆伟, 柴旭东. 一种新的仿真组件模型及其实现[J]. 南京航空航天大学学报, 2006,38(6): 781-784.
- [34] 胡斌, 常青. 军事概念模型建模实践分析与研究[J]. 系统仿真学报, 2008,12(20): 3085-3088.
- [35] Balci O. Ormsby W F. Conceptual Modeling for Designing Large-scale Simulations [J]. Journal of Simulation. 2007, 1(3): 175-186.
- [36] 余滨, 段采宇, 饶德虎. C4ISR 系统军事需求描述概念模型[J]. 国防科技大学学报, 2008,2(30): 112-118.
- [37] 陈静杰, 邵丽娜, 耿宏. 基于仿真组件模型(SCM)的航空电子组件仿真模型研究[J]. 科学技术与工程, 2016,10(16): 50-56.
- [38] 卞春丽, 岳增坤, 陈炜. 仿真模型基于 BOM 的组件化设计[J]. 舰船电子工程, 2011, 1(31):109-114.
- [39] 张继春, 吴伟泽. 美国陆军数字化部队研究现状和发展趋势[J], 舰船电子工程, 2013.3.
- [40] 王迎春, 席学强, 刘兵. 装备指挥训练模拟系统中仿真模型体系研究[J]. 装备学院学报, 2012,8(4): 97-101.
- [41] 杨晓段, 李元左, 刘曙云. 空间军事系统复杂问题求解的多元化模型体系[J]. 兵工自动化, 2009,7(28): 49-53.
- [42] 叶丰, 蔡业泉, 刑继娟. 武器装备体系对抗仿真模型开发关键技术研究[J]. 军事运筹与系统工程, 2008,12(4): 63-68.
- [43] 余文广, 王维平, 李群. 模型驱动的组件化 AGENT 仿真模型开发方法[J]. 系统工程与电子技术, 2011,8(33): 1907-1911.
- [44] 卿杜政, 李伯虎, 等. 基于组件的一体化建模仿真环境(CISE)研究[J]. 系统仿真学报, 2008, 20(4): 900-903.
- [45] 杨兴涛, 苏桂平, 王瑞芳. 模型驱动的构件组装的研究与实现[J]. 计算机系统应用, 2009,4:96-100.
- [46] 赵倩, 毕长剑, 吉宁. 基于网络的军事模型服务体系结构[J]. 系统仿真学报, 2006,8: 3520-3529.
- [47] 葛冰峰, 陈英武, 王军民. 基于功能的武器装备体系结构描述方法[J]. 系

- 统工程与电子技术, 2010,1(32): 95-100.
- [48] 王凯, 吴小良. 数字化战场指挥控制系统[J]. 火力与指挥控制, 2008,33(1):4-8.
- [49] 李宏权, 邓桂龙. 战役训练模型服务体系与技术方法[M]. 北京: 国防工业出版社, 2012.
- [50] 艾利锋, 刘春煌, 杜彦华. 5T 系统整合中基于模型驱动的 Web 服务组合方法研究[J]. 中国铁道科学, 2007,4 (28): 118-124.
- [51] 曾星宇. 基于 HTN 规划的服务组合框架的研究与实现[D]. 上海: 上海交通大学软件学院, 2012,12: 42-52.
- [52] 刘式宋, 刑利华. 基于精确指挥的数字化部队战斗力生成[J], 四川兵工学报, 2009,30(10):130-134.
- [53] 周东详. 多层次仿真模型组合理论与集成方法研究[D]. 长沙: 国防科学技术大学研究生院, 2007,10: 85-95.
- [54] 王志刚, 刘虎. 解读美军数字化部队和目标部队的区别[J], 国防科技, 2004.10.
- [55] 贾丽, 张和明. 面向服务的分布式建模仿真框架研究[J]. 系统仿真学报, 2007,19 (20): 4680-4683.
- [56] 李鑫, 程渤, 杨国纬. 一种基于事件的 Web 服务组合方法[J]. 软件学报, 2009,12 (20): 3102-3108.
- [57] 雷永林. 仿真模型重用理论与异构集成技术研究[D]. 长沙: 国防科学技术大学研究生院, 2006,9: 25-35.
- [58] 温嘉佳. Web 服务组合及其相关技术的研究[D]. 北京: 北京邮电大学, 2006,9: 65-78.
- [59] 刘思培, 刘大有, 齐红. 基于服务组链的 Web 服务组合方法[J]. 吉林大学学报(工学版), 2010,1 (40): 149-155.
- [60] 孔维梁, 刘清堂, 杨宗凯, 等. 基于动态 QoS 的 Web 服务组合[J]. 计算机科学, 2012,2 (39): 268-272.
- [61] 宋莉莉. 基于 SOA 的建模与仿真框架及仿真服务发现技术研究[D]. 长沙: 国防科学技术大学研究生院, 2009,3: 29-47.
- [62] 王鸿洁, 常国岑, 李学军. 军事通用仿真模型及其集成框架研究[J]. 系统仿真学报, 2007, 19(3): 486-490.

- [63] 王映辉. 软件构件体系结构——原理、方法与技术[M]. 北京: 机械工业出版社, 2009.
- [64] 王维平, 雷永林, 朱一凡. 基于元模型的可重用仿真模型表示方法研究[J]. 计算机仿真, 2007, 24(8): 93-97.
- [65] 刘忠, 张维明, 阳东升, 等. 作战计划系统技术[M]. 北京: 国防工业出版社, 2007.
- [66] 吴延林. 仿真模型库系统的研究与实现[D]. 长沙: 国防科学技术大学, 2005.
- [67] 吴延林, 邱晓刚, 刘宝宏. 基于 Web 仿真模型库系统的总体框架[J]. 兵工自动化, 2006,(01).
- [68] 张振兴. 基于 Web 服务的模型库系统的研究与实现[J]. 华北电力大学学报, 2003,(04).
- [69] 柳寒冰, 宿红毅, 张晗. 军用仿真中基于 Web CMS 的仿真资源管理机制研究[J]. 北京工业大学学报, 2010,(01).
- [70] 杜强芳, 杨明, 王子才. 建模与仿真资源库系统的初探[J]. 系统仿真学报, 2003,(02).
- [71] 黄卓. 装备综合保障信息系统模型管理技术研究[D]. 长沙: 国防科学技术大学, 2003.
- [72] 张文苑. 虚拟样机中飞行仿真模型库及建模环境的研究[D]. 北京: 北京航空航天大学研究生院, 2003,3.
- [73] 肖芳雄. 面向 QoS 的 Web 服务组合建模和验证研究[D]. 南京: 南京航空航天大学研究生院, 2010,9:34-48.
- [74] 冯晓宁, 王卓, 顾磊, 等. 行为驱动的仿真模型组合方法研究[J]. 系统仿真学报, 2012, 1(24): 20-26.
- [75] 李群, 雷永林, 侯洪涛, 等. 仿真模型可移植性规范及其应用[M]. 北京: 电子工业出版社, 2010.
- [76] 刘东玉, 唐忠, 邱超, 等. 雷达电子战仿真模型库构建方法研究[J]. 舰船电子工程, 2009(05).
- [77] 郑晓薇, 逯文晖. 分布决策模型库目录管理系统设计与实现[J]. 计算机工程与设计, 2010(01).
- [78] 黄柯棣, 邱晓刚, 等. 建模与仿真技术[M]. 长沙: 国防科技大学出版社, 2011.

